

Institutionen för Informatik
Handelshögskolan
Göteborgs Universitet



Användarna – Nyckeln till ökad användbarhet

Författare: Johanna Palm
Annika Östman
Handledare: Jan Ljungberg

Magisteruppsats 20 poäng
Vårterminen 2001

Abstrakt. Målet med den här uppsatsen är att visa hur viktigt det är att ta med användarna i systemutvecklingen för att få ett datorsystem med hög användbarhet. Vi beskriver i uppsatsen hur vi har gått tillväga för att förbättra användbarheten i ett befintligt system genom att integrera användare. Detta tillvägagångssätt har vi uttryckt som en metod som vi anser att man skall kunna tillämpa på de flesta typer av system som kännetecknas av låg användbarhet. Metoden har främst tagits fram utifrån olika utvärderingsmetoder som beskrivs i teorin. Vi har samlat in empiri till uppsatsen genom i första hand etnografiska metoder, såsom observation och intervju. Vi började med att utvärdera ett befintligt system som vi använt som praktikfall i denna uppsats. Genom utvärderingen fick vi veta vad användarna tyckte om systemet och vilka problem som de upplevde med dess användargränssnitt. Därefter tog vi fram en prototyp av användargränssnittet tillsammans med användarna. Vi utvärderade sedan denna prototyp för att se om användbarheten ökat i systemet. Resultatet från undersökningen visade att man, med hjälp av vår metod, i hög grad kan förbättra användbarheten i ett systems användargränssnitt genom att integrera användaren.

Tack!

Vi vill börja med att tacka våra handledare på Mind AB i Göteborg, Tale Gulliksen och Åsa Tilly. Tale för hon var den som tog initiativet till uppsatsens ämne och hjälpte oss att komma igång med vårt uppsatsskrivande. Åsa för all uppmuntring och positiv feedback samt för hennes enorma engagemang.

Vi vill även tacka Mikael Lancehielm, teknisk konsultchef på Mind AB, för hans positiva bemötande vilket gjorde att vi genast kände oss välkomna på företaget.

Vidare vill vi tacka Karin Eklund och Peter Nilsson som med stor entusiasm delade med sig av sina expertkunskaper inom människa-dator-interaktion. Deras erfarenheter var av oerhörd stor hjälp för oss.

Vi vill även tacka alla på Mind som ställde upp på våra användbarhetstester, samt de andra på företaget som stöttade och intresserade sig för vårt arbete.

Slutligen vill vi tacka vår handledare, Jan Ljungberg på institutionen för Informatik samt hans kollega Magnus Bergquist för värdefulla råd och synpunkter.

INNEHÅLLSFÖRTECKNING

DEL I INLEDNING5

1. INTRODUKTION5

1.1 PROBLEMBAKGRUND	5
1.2 PROBLEMFORMULERING	6
1.3 SYFTE OCH FÖRVÄNTAT RESULTAT.....	6
1.4 AVGRÄNSNING	7
1.5 MÅLGRUPP	7
1.6 GRUNDLÄGGANDE BEGREPP	7
1.7 DISPOSITION	8

2. FORSKNINGSMETOD9

2.1 FORSKNINGSMETODER UTIFRÅN TEORIN	9
2.1.1 <i>Etnografi</i>	9
2.1.2 <i>Observation</i>	9
2.1.3 <i>Intervju</i>	10
2.1.4 <i>Enkät</i>	11
2.2 VÅR FORSKNINGSMETOD	11
2.2.1 <i>Den studerade verksamheten</i>	11
2.2.2 <i>Vårt urval</i>	11
2.2.3 <i>Vald forskningsmetod</i>	13

DEL II TEORI.....17

3. ANVÄNDARCENTRERAD DESIGN17

4. ANVÄNDARGRÄNSSNITTSDESIGN18

4.1 RIKTLINJER FÖR ANVÄNDARGRÄNSSNITTSDESIGN	19
4.1.1 <i>Fönster och Skärmdesign</i>	20
4.1.2 <i>Menyer</i>	21
4.1.3 <i>Formulär</i>	23
4.1.4 <i>Systemmeddelanden</i>	23
4.1.5 <i>Färger</i>	24

5. UTVÄRDERINGSMETODER.....25

5.1 VAL AV UTVÄRDERINGSMETOD.....	26
5.2 INSPEKTION.....	29
5.2.1 <i>Heuristisk Evaluering</i>	29
5.2.2 <i>Standardinspektion</i>	32
5.3 ANVÄNDARTEST.....	32
5.3.1 <i>Tänka-högt-protokoll</i>	33
5.3.2 <i>Samupptäckar-metoden</i>	33
5.3.3 <i>Coaching Metoden</i>	33
5.4 EMPIRISK PRÖVNING.....	34
5.4.1 <i>Fältobservation</i>	34
5.4.2 <i>Fokusgrupper</i>	34
5.4.3 <i>Intervju</i>	35
5.4.4 <i>Enkät</i>	35
5.5 PROTOTYPING.....	35

DEL III RESULTAT38

6. RAMVERK38

7. EMPIRI	39
7.1 UNDERSÖKNING 1	39
7.1.1 Syfte	39
7.1.2 Metod.....	40
7.1.3 Arbetsprocedur.....	40
7.1.4 Evaluering utifrån de tio tumreglerna.....	41
7.1.5 Resultat från undersökning 1.....	44
7.2 UNDERSÖKNING 2	44
7.2.1 Syfte	44
7.2.2 Metod.....	45
7.2.3 Testomgivningen och testutrustningen.....	45
7.2.4 Testledarens och observatörens roll	46
7.2.5 Testprocedur	47
7.2.6 Resultat från undersökning 2.....	48
7.3 UNDERSÖKNING 3	50
7.3.1 Syfte	50
7.3.2 Metod.....	50
7.3.3 Testomgivningen och testutrustningen.....	51
7.3.4 Testledarnas roller.....	52
7.3.5 Testprocedur	52
7.3.6 Resultat från undersökning 3.....	53
7.4 UNDERSÖKNING 4	58
7.4.1 Syfte	58
7.4.2 Metod.....	58
7.4.3 Testomgivningen och testutrustningen.....	59
7.4.4 Testledarens och observatörens roll	60
7.4.5 Testprocedur	60
7.4.6 Resultat från undersökning 4.....	61
8. DISKUSSION.....	63
8.1 SLUTLIGT RESULTAT	63
8.2 REFLEKTION	67
9. SLUTSATS	69
LITTERATURFÖRTECKNING	70
BILAGOR	72
BILAGA A: UNDERSÖKNING 1	72
BILAGA B: UNDERSÖKNING 2	73
BILAGA C: UNDERSÖKNING 3	76
BILAGA D: UNDERSÖKNING 4.....	81

Del I Inledning

1. Introduktion

Vi har i vår magisteruppsats undersökt om och i så fall hur man kan förbättra användbarheten i ett befintligt systems användargränssnitt genom att i enlighet med teorin utföra undersökningen på ett internetföretag.

Magisteruppsatsen är skriven på internetföretaget Mind AB i Göteborg. Företagets anställda redovisar idag sin arbetstid i ett standardaffärssystem som är uppbyggt både som en klient och som en webbapplikation. Klienten används bl.a. för tidsredovisning, utvärdering och fakturering. Webbapplikationen används enbart för tidsredovisning. Missnöjet med systemets användbarhet är stort hos företagets anställda, vilket har medfört att det finns ett motstånd att använda sig av det. Det finns därför en önskan på Mind att det ska ske en förbättring av systemets användargränssnitt.

1.1 Problembakgrund

Människa-dator-interaktion (MDI), eller engelskans Human-computer-interaction (HCI), är ett begrepp som blivit allt vanligare inom IT-branschen. Med detta begrepp avser man design av datorsystem som är säkra, effektiva, enkla och trevliga att använda såväl som funktionella. MDI handlar om att förstå hur människor använder datorsystem så att man kan designa bättre system som tillfredsställer användarnas behov¹.

Det talas och skrivs alltmer om hur viktigt det är att ta med användaren i systemutvecklingen för att öka användbarheten, tyvärr följs detta inte alltid i praktiken. Vid tidsbrist och begränsade ekonomiska resurser måste man dra ned på något och risken är att det är användbarhetsarbetet i systemutvecklingen som blir lidande. Det finns alltför många system idag som överhuvudtaget inte används. En viktig orsak till detta är, enligt en studie av Lederer och Prasad², ett dåligt samarbete mellan användare och systemutvecklare. Detta leder till brister i systemets användbarhet och skapar motstånd hos användarna att använda systemet. Användarna får ofta ett färdigt system som de kanske inte har efterfrågat och som de inte heller har varit med och tagit fram. Detta gör att de känner en motvilja att använda det.

Allt fler inser vikten av att utveckla system tillsammans med användarna för att ta fram ett system som tillfredsställer användarnas behov. Ett huvudsyfte med MDI är något som kallas Användarcentrerad design, eller User-centred-design (UCD), där man fokuserar på människor, deras arbete och deras miljö, och hur teknologin bäst kan utvecklas och designas för att stödja dem³. Enligt användarcentrerad design kretsar all utveckling kring att användaren är i centrum.

Man bör under systemutvecklingsprocessen utvärdera systemet för att ta reda på vad användarna vill ha och vilka problem som finns. Ju större förståelse designern har om användarnas behov, desto bättre kommer produkten att bli⁴.

¹ Preece, Jenny, *A guide to usability: Human factors in computing* (1993)

² Flynn, Donald, *Information System Requirements: Determination & Analysis* (1997)

³ Preece, Jenny, *Human-computer-interaction* (1996)

⁴ Ibid

De flesta systemutvecklingsmetoder i litteraturen som beskriver UCD utgår ifrån att man utvecklar system från början, dvs allt ifrån det att kunden gör en förfrågan på ett system till dess att systemet realiseras. Vi saknar dock en metod som beskriver hur man skall utveckla ett *befintligt* system tillsammans med användaren för att nå ökad användbarhet. Hur skall man göra med alla de system som redan existerar men som kännetecknas av brister i användbarheten? Många av dessa system används inte trots att det i många fall ligger mycket tid, pengar och stort arbete bakom att ta fram dem.

Genom att öka ett systems användbarhet skulle man kunna minska missnöjet med systemet vilket skulle innebära större och bättre användning av det. Då vi saknar en metod för hur man förbättrar användbarheten i ett befintligt system ser vi det som en utmaning att försöka ta fram en sådan.

På Mind har man i över ett år använt sig av ett affärssystem där alla anställda varje vecka måste redovisa sin arbetstid. Då vi inte vill namnge detta affärssystem har vi valt att i fortsättningen kalla det "Systemet". Missnöjet har varit väldigt stort med detta system vad gäller användbarheten. Man anser det vara svårt och bökigt att använda för en, kan tyckas, så enkel uppgift som att redovisa sin arbetstid.

I och med detta problem fick vi en förfrågan från Mind att försöka ta fram ett alternativt användargränssnitt till det system som används idag för att på så sätt förbättra användbarheten. Vi såg detta system som ett utmärkt praktikfall för att *ta fram en metod för att öka användbarheten hos ett befintligt system* då vi idag saknar en sådan. Det ger oss också ett utmärkt tillfälle att lära oss att utveckla ett användargränssnitt tillsammans med användare och samtidigt få erfarenhet för hur man arbetar med dessa frågor i en verksamhet.

På grund av den dåliga användbarheten i Systemet förutsätter vi att ingen tid, eller en mycket begränsad tid, har lagts ner på design av ett användbart användargränssnitt. Detta trots att Systemet har funnits på marknaden i mer än tio år och att det kontinuerligt kommer ut nya versioner.

1.2 Problemformulering

Hur ökar man användbarheten i ett befintligt system?

- Kan man genom att integrera användare i användargränssnittsdesignen öka användbarheten i ett system?
- Hur skall den metod se ut som ligger till grund för utvecklingen av användargränssnittet i ett befintligt system?

1.3 Syfte och förväntat resultat

Syftet med uppsatsen är att:

- skapa en metod för användargränssnittsdesign i ett befintligt system som härleds från teorin och verifieras i verkligheten.
- utvärdera ett specifikt användargränssnitt med och utan användare i enlighet med metoden i första punkten.
- ta fram ett prototypförslag på förbättrat användargränssnitt utifrån utvärderingen.

Vårt förväntade resultat är att systemets användbarhet ökar om användare är med och bestämmer utformningen.

1.4 Avgränsning

För att avgränsa arbetet har vi valt att koncentrera oss på användargränssnittet i webbapplikationen och därmed endast tidsredovisningsdelen. För att ytterligare avgränsa arbetet har vi valt att inte skapa ett fullt fungerande system med bakomliggande funktionalitet, utan endast göra en prototyp av användargränssnittet. Vidare kommer vi, när det gäller förbättringen av användargränssnittet, avgränsa oss till att endast göra ändringar inom några få områden och därmed fokusera på de största och allvarligaste användbarhetsproblemen i Systemet.

1.5 Målgrupp

Den här uppsatsen vänder sig till alla de som har system som kännetecknas av dålig användbarhet och som därför behöver förbättras. Den vänder sig till studenter som studerar på någon typ av IT-utbildning och till de som arbetar inom IT-området i yrkeslivet. Då uppsatsen i första hand behandlar frågor som rör design av användargränssnitt är den kanske av störst intresse för de personer som studerar eller arbetar inom detta område. Vi anser dock att även de som är mer tekniskt inriktade borde läsa denna uppsats för att få större förståelse för betydelsen av att ta med användaren i systemutvecklingen.

1.6 Grundläggande begrepp

För att skapa en gemensam begreppsbild bland läsarna har vi valt att förklara de begrepp som vi använder i uppsatsen.

Ett begrepp som används flitigt i uppsatsen är ”*användbarhet*”. Enligt Rubin⁵ finns ingen definition på användbarhet som är accepterad av alla MDI-forskare inom användbarhetsområdet. Vi har valt att följa Jacob Nielsens⁶ förklaring på begreppet. Han definierar användbarhet utifrån fem punkter:

- *Lätt att lära*: systemet skall vara utformat så att användaren snabbt kan börja utföra sina uppgifter.
- *Effektivt*: systemet skall vara effektivt att använda så att användarna kan nå en hög produktivitetsnivå när de lärt sig systemet.
- *Lätt att minnas*: det skall vara lätt att komma ihåg var saker och ting finns i systemet utan att behöva lära om på nytt.
- *Felhantering*: systemet skall vara designat så att användaren har en låg felfrekvens, vilket innebär att det skall vara svårt för användaren att göra fel i systemet. Gör användaren ändå fel skall felet vara lätt att åtgärda.
- *Tillfredsställelse*: systemet skall kännas behagligt att använda.

”*Användbarhetsproblem*” är en annan term som återkommer i uppsatsen. Användbarhetsproblem är aspekter i systemet som bidrar till att användbarheten försämrats. Utgår man från Nielsens definition på användbarhet innebär system med användbarhetsproblem system som som *inte* följer ovanstående punkter.

⁵ Rubin, Jeffrey, *Handbook of usability testing* (1994)

⁶ Nielsen, Jakob, *Usability engineering* (1993)

Vidare har vi begreppet ”*användargränssnitt*”. Användargränssnitt eller gränssnitt, som vi ibland har valt att kalla det, är det som finns mellan användaren och systemet. Det är användargränssnittet som användaren ser när de tittar på systemet och det är med hjälp av användargränssnittet som användaren kan interagera med systemet⁷.

Ett annat begrepp som är relaterat till användargränssnitt är ”*fönster*”. Fönster är en del av ett användargränssnitt. Vanligtvis är fönstret rektangulärt. Ett fönster kan vara så stort att det täcker hela skärmen eller så kan ett fönster vara ett kort meddelande i form av ett litet fönster som ligger ovanpå ett annat fönster⁸.

1.7 Disposition

Uppsatsens dispositionen ser ut på följande sätt:

Kapitel 2, Forskningsmetod:	Här beskrivs några forskningsmetoder utifrån teorin. Kapitlet fortsätter med en beskrivning av vår valda metod och hur vårt urval gått till.
-----------------------------	--

Del II Teori

Kapitel 3, Användarcentrerad design:	Här tar vi upp begreppet användarcentrerad design. Vi förklarar vad begreppet innebär och hur man kan använda det i systemutveckling.
--------------------------------------	---

Kapitel 4, Användargränssnittsdesign:	I detta kapitel tas ett antal riktlinjer upp för att designa ett användbart användargränssnitt. Vi har begränsat oss till att endast beskriva de riktlinjer som är relevanta för vår studie.
---------------------------------------	--

Kapitel 5, Utvärderingsmetoder:	Här beskriver vi några av de utvärderingsmetoder som finns att välja bland när man skall utvärdera ett datorsystem.
---------------------------------	---

Del III Resultat

Kapitel 6, Ramverk:	Här beskriver vi den metod som vi tagit fram enligt teorin.
---------------------	---

Kapitel 7, Empiri:	I detta kapitlet verifierar vi den framtagna metoden. Vi beskriver den data vi har samlat in för att kunna svara på vår problemformulering. Kapitlet är uppdelat på de fyra undersökningar som har gjorts.
--------------------	--

Kapitel 8, Diskussion:	Detta avsnitt består av två kapitel, Slutligt resultat och Reflektion. Vi
------------------------	---

⁷ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁸ Galitz, Wilbert, *The essential guide to user interface design* (1996)

diskuterar här resultaten utifrån problemställningen respektive gör reflektioner kring vårt arbete.

2. Forskningsmetod

I detta avsnitt beskriver vi utifrån litteraturen ett antal forskningsmetoder som vi anser vara relevanta för vår problemställning. Vi beskriver därefter på vilket sätt vi har använt oss av dessa metoder i vår studie.

2.1 Forskningsmetoder utifrån teorin

Man delar ofta in forskning i antingen kvantitativ- eller kvalitativ forskning. Kvantitativ forskning innebär att man använder sig av statistiska bearbetnings- och analysmetoder. Den kvalitativa forskningen innebär att man använder en mer verbal analysmetod. Vilken typ av metod man väljer beror på syftet med studien och hur frågeställningen ser ut som man vill ha svar på. Vill man exempelvis ha svar på frågor som t ex Var? Hur? Vilka är skillnaderna? Vilka är relationerna? bör man använda en kvantitativ metod. Vill man tolka och förstå människor eller vill ha svar på frågor som t ex Vad är detta? Vilka är de underliggande mönstren? bör man använda en kvalitativ metod⁹.

2.1.1 Etnografi

Etnografi är en kvalitativ forskningsmetod där forskaren försöker bli en del av den verksamhet som undersöks för att på så sätt förstå beteendet hos de som studeras¹⁰. Avsikten med en etnografisk studie är att se sociala handlingar som utförs av medlemmar i en organisation. Detta ger möjligheten att se hur medlemmarna i en organisation förstår och utför sitt arbete i vardagen. Fördelen med etnografisk studie är att den synliggör det sociala livet i en verklig arbetssituation¹¹.

Vid systemutveckling skapar den etnografiska metoden förutsättningar för att få grepp om det verkliga arbetslivet där datasystemet faktiskt skall användas och fungera. Detta gör att systemdesigners lättare kan få ett användarperspektiv i sin systemdesign¹². Nackdelar med forskningsmetoden är att den ofta uppfattas som osystematisk och ostrukturerad¹³. Andra nackdelar är enligt Hughes att det tar lång tid att genomföra den och att det är svårt att göra studien i stor skala.

2.1.2 Observation

Observationsmetoden är en av de viktigaste metoderna inom etnografi och är framför allt en kvalitativ metod som används när man vill samla information inom områden som berör beteenden och skeenden i naturliga situationer. Man vill studera vad människor gör istället för vad de säger att de gör. Man studerar inte bara fysiska handlingar utan även verbala yttranden, relationer mellan individer, känslouttryck etc. En observation kan antingen vara deltagande eller så kan forskaren agera som "flugan på väggen". Den deltagande observationen innebär att observatören observerar verksamheten samtidigt som denne deltar i den verksamhet som observeras. Att agera

⁹ Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning* (1991)

¹⁰ Easterby-Smith, Mark et al., *Management Research: An introduction* (1991)

¹¹ Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design* (1994)

¹² Blomberg, J et al., *Ethnographic Field Methods and Their Relation to Design Participatory Design* (1993)

¹³ Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design* (1994)

"flugan på väggen" innebär däremot att man observerar en verksamhet utan att vara en del av den.

En observation kan vara strukturerad eller ostrukturerad. En strukturerad observation innebär att man i förväg bestämt vilka fenomen man skall fokusera på i observationen. En ostrukturerad observation används när man vill erhålla så mycket kunskap som möjligt dvs man observerar allt i en speciell omgivning.

Fördelen med observationsmetoden är att undersökningspersonerna inte behöver ha en tydlig minnesbild som de sedan skall vidarebefordra till de som utför undersökningen så att de uppfattar den rätt. En annan fördel är att det krävs mindre samarbete med de utvalda individerna till skillnad från de flesta andra metoder. Nackdelen med metoden är att den är tidskrävande. Vidare diskuteras aspekterna på observation som forskningsteknik. När fångar vi ett spontant beteende? När vet vi att de beteenden som observeras är representativa?¹⁴.

2.1.3 Intervju

Intervju är en datainsamlingsmetod som är viktig i många metodansatser, däribland etnografi. En intervju kan vara strukturerad, semistrukturerad eller ostrukturerad, gränserna mellan dessa är dock suddiga. En strukturerad intervju är baserad på noggrant förberedda frågor där intervjuaren frågar samma frågor till varje person som intervjuas och, om det är möjligt, i samma tonfall. Ofta ställs frågorna i samma ordning¹⁵. I en strukturerad intervju lämnas lite utrymme åt den intervjuade personen att svara inom och intervjuaren kan förutsäga vilka svarsalternativ som är möjliga¹⁶. Den enklaste typen av strukturerad intervju är den där svaren är korta och där intervjuaren snabbt kan bocka för svarsalternativet. Denna typ av intervju används oftast i kvantitativa undersökningar som exempelvis marknadsundersökningar i ett köpcentrum där man undersöker attityden till en viss produkt. Det är dock ett mödosamt arbete, p.g.a. att man behöver ett stort antal svar för att få ett trovärdigt resultat¹⁷.

Det är vanligare att intervjuer är mer kvalitativa med en lägre grad av strukturerade frågor. I en semistrukturerad intervju har man liksom i en strukturerad intervju, väl förberedda frågor. Man är dock inte lika noga med att ställa frågorna i en viss ordning utan anpassar frågorna efter de svar man får. Frågorna är mer öppna vilket innebär att den intervjuade kan svara mer fritt och man försöker likna intervjun vid ett samtal. Skillnaden mellan en semistrukturerad och ostrukturerad intervju är marginell. Man kan dock säga att en helt ostrukturerad intervju innebär att intervjupersonen får tala helt fritt utan någon större påverkan från intervjuaren. Denna typ av intervju rekommenderas dock inte då den kan vara svår att sammanställa. En semistrukturerad, eller ostrukturerad, fråga skulle t ex kunna vara; "Vad skulle du göra om du vann en miljon?" Denna typ av intervju används när man vill få en mer grundläggande förståelse för hur individer uppfattar en viss situation¹⁸.

Fördelen med intervjumetoden är att man får mer information än endast svaret på frågan. Intervjupersonen använder inte alltid ord för att uttrycka sig, utan också gester

¹⁴ Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning* (1991)

¹⁵ Easterby-Smith, Mark et al., *Management Research: An introduction* (1991)

¹⁶ Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning* (1991)

¹⁷ Easterby-Smith, Mark et al., *Management Research: An introduction* (1991)

¹⁸ Ibid

och ansiktsuttryck, som kan vara användbar information. Nackdelen med metoden är att det kan ta lång tid att utföra intervjuer och framför allt att sammanställa svaren, särskilt om intervjun är av en mer ostrukturerad karaktär¹⁹.

2.1.4 Enkät

Liksom intervju bygger denna metod på att man vill samla in information med hjälp av frågor. Enkätundersökningar är oftast kvantitativa där man ställer frågor till en viss målgrupp i samhället. En enkät kan, liksom en intervju, bestå av dels strukturerade frågor med färdiga svarsalternativ och dels av öppna frågor. Vanligtvis brukar en enkät ha en hög grad av strukturerade frågor där man får välja mellan ett antal svarsalternativ²⁰. Fördelen med metoden är att man relativt billigt kan nå ut till många människor. Vidare finns det idag många bra och effektiva statistikprogram som underlättar sammanställningen av svaren²¹. En annan fördel med metoden är att den inte är så tidskrävande. Det går relativt snabbt för undersökningspersonerna att svara på en enkät, särskilt om frågorna är strukturerade med ett antal svarsalternativ. Nackdelen med metoden är att en enkät endast ger en begränsad bild av människors synpunkter och erfarenheter. Det krävs också att frågorna är genomtänkta och noggrant formulerade så att det inte sker något missförstånd. En enkät kan även med fördel användas som komplement i en etnografisk studie²².

2.2 Vår forskningsmetod

I det här avsnittet kommer vi förklara hur vi har gått tillväga i vår undersökning. Vi börjar med att beskriva den verksamhet och dess användare som ligger till grund för undersökningen. Detta gör vi för att öka förståelsen för vårt urval samt den verksamhet vi studerar. Vi beskriver därefter hur urvalet gått till samt ger en övergripande beskrivning av de empiriska undersökningar vi gjort.

2.2.1 Den studerade verksamheten

Då Mind är ett IT-företag där alla anställda dagligen arbetar vid en dator är erfarenheten av datorarbete mycket hög.

Företaget består i huvudsak av fyra enheter; Human & Process, Design & Communication, Technology och Central. Dessa enheter är av olika storlek och datakunskaperna skiljer sig mellan de olika enheterna beroende på vilket sätt man använder datorn. De största enheterna, Technology och Design & Communication, har nästan dubbelt så många anställda som de övriga enheterna.

Skillnaden i användarnas kunskap vad gäller Systemet beror dels på hur länge de har använt sig av Systemet och dels på vilket sätt de använder sig av det. På vilket sätt de använder Systemet beror i första hand på vilken enhet de arbetar på och vilken ansvarsnivå de befinner sig på.

2.2.2 Vårt urval

Målgruppen för undersökningen är de personer som måste redovisa sin arbetstid i Systemet. Urvalet skall i största möjliga mån reflektera målgruppen.

¹⁹ Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning* (1991)

²⁰ Ibid

²¹ Easterby-Smith, Mark et al., *Management Research: An introduction* (1991)

²² Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning* (1991)

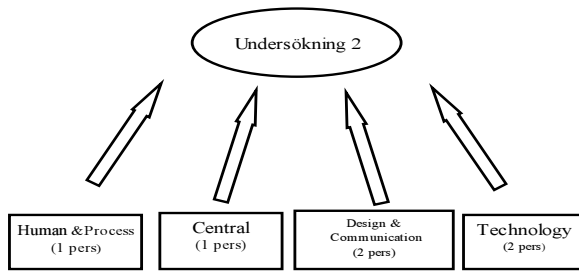
Vårt urval har baserats på ett mail vi skickade ut till målgruppen där vi frågade vilka som kunde tänka sig att ställa upp i undersökningen. Frågan ställdes p.g.a. att de anställda är mycket upptagna och att det därför är svårt att slumpmässigt välja ut personer från hela företaget till undersökningen. I mailet frågade vi även vad de tyckte om Systemet. Vi fick därmed en känsla för vilka problem de anställda upplevde med Systemet. Mailet skickades ut alla anställda på Mind, dvs ca 110 stycken. Av dessa fick vi 74 svar varav 20 svarade att de kunde tänka sig att ställa upp i undersökningen.

Vi har gjort sammanlagt fyra undersökningar i denna studie varav tre har gjorts med undersökningspersoner och en har gjorts utan. Ingen av undersökningspersonerna har deltagit i mer än en undersökning. Den sista undersökningen med användare gjordes av praktiska skäl inte på Mind, utan på ett annat företag som även de använde Systemet för tidsredovisning²³.

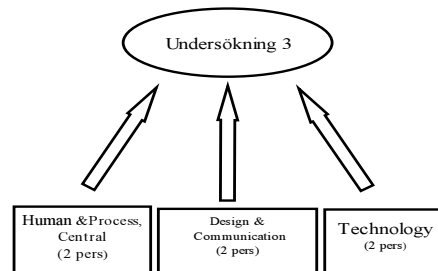
Då den första undersökningen inte inkluderade några användare kommer vi inte att beskriva något urvalsförfarande för denna undersökning. I den andra och den tredje undersökningen använde vi oss av en blandning av två urvalsmetoder. Dels gjorde vi ett stratifierat urval där vi delat upp populationen i olika strata (strata = kategori eller grupp)²⁴ och dels är vårt urval ett "självselektionsurval", vilket innebär att de som vill vara med i undersökningen fick svara "ja" på mailet vi skickade ut. I den andra undersökningen utgörs stratan av de fyra enheterna som finns på Mind medan vi i undersökning 3 valde att slå ihop två enheter då vi här ansåg att det räckte med tre grupper. Bland de "ja-svar" vi fick på mailet valdes sex personer ut till vardera undersökning. Den sista undersökningen, som gjordes på ett annat företag, var urvalet baserat på självselektion. I denna sista undersökning deltog fyra personer. Figureerna nedan visar urvalet till de tre undersökningarna som involverade användare.

²³ Mind AB i Göteborg lade ner sin verksamhet innan vi hann göra den sista undersökningen

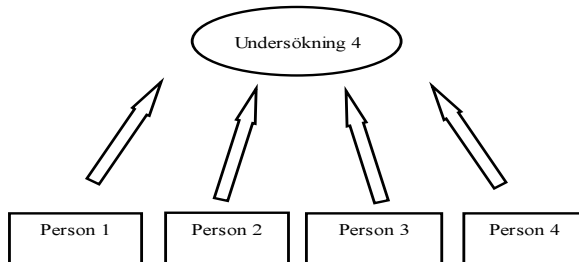
²⁴ Undheim, Johan Olav, *Statistik från ord till formel* (1988)



Figur 2.1. Urval till undersökning 2



Figur 2.2. Urval till undersökning 3



Figur 2.3. Urval till undersökning 4

2.2.3 Vald forskningsmetod

Vi har valt att använda ett flertal metoder för att få svar på vår frågeställning. Anledningen till detta är att vi genom att blanda olika metoder kan få olika typ av information som sedan kan kopplas samman för att öka validiteten och tillförlitlighet i vår uppsats.

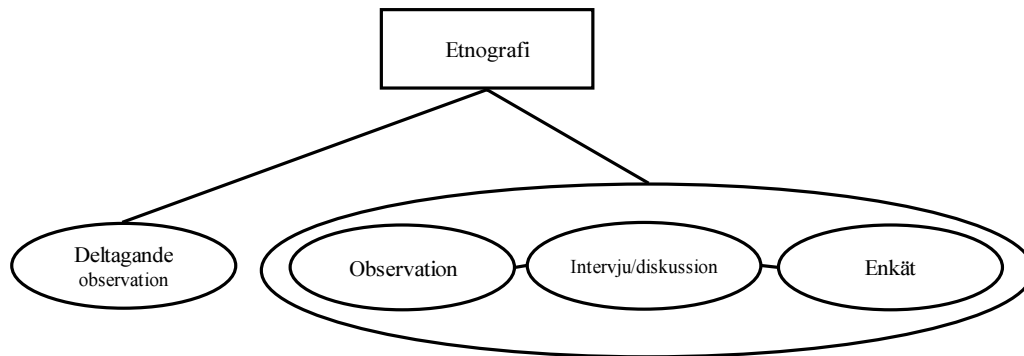
Som nämndes i föregående avsnitt har vi gjort fyra undersökningar. I de två första undersökningarna utvärderade vi användargränssnittet i det befintliga tidsredovisningssystemet. I den tredje undersökningen utarbetade vi en prototyp av det nya användargränssnittet och slutligen i den sista undersökningen, utvärderade vi den framtagna användargränssnittsprototypen.

I detta forskningsmetodavsnitt kommer vi övergripande beskriva på vilket sätt vi har använt oss av de olika forskningsmetoderna. Senare i resultatdelen gör vi en mer ingående beskrivning över hur vi kombinerat forskningsmetoder med utvärderingsmetoder för att utvärdera användbarheten hos ett datorsystem.

När vi började undersöka hur vi skulle gå tillväga för att förbättra användargränssnittet hos ett befintligt system insåg vi vikten av att arbeta tillsammans med användarna. Ett sätt att göra detta på är med hjälp av en etnografisk metod. Därför genomsyras vår arbetsprocess av en etnografisk ansats där vi använt oss av deltagande observation, observation, intervju och enkät (se figur 2.4 nedan). Den etnografiska ansatsen har bl.a. tagit sig uttryck i korta etnografiska studier av typ etnografisk observation, som beskrivs av Shneiderman²⁵, samt Hughes²⁶ "quick and dirty ethnography". Dessa förklaras närmare i samband med respektive undersökning nedan.

²⁵ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

²⁶ Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design* (1994)



Figur 2.4. Beståndsdelarna i vår etnografiska studie

Figur 2.4 visar de metoder vi använt oss av inom ramen för vår etnografiska studie. Anledningen till att observation, intervju/diskussion och enkät är sammankopplade i figuren ovan är att vi använde oss av dessa metoder i samband med varandra. Detta beskrivs mer ingående nedan.

Vi utförde en deltagande observationen genom att vi under ungefär sex månader tillbringade vår tid på Mind i Göteborg och på det sättet blev en del av företaget. Vi fikade, åt lunch och pratade med de anställda under denna period. Under dessa informella möten diskuterades bl.a. Systemet. Vi fick på det sättet information om vad de anställda tyckte om Systemet, vad som var dåligt, bra och vad de hade för förslag på förbättringar. Vi hade från början inte som avsikt att använda en deltagande observation i denna studie, utan detta var något som växte fram under tiden vi var på företaget.

Den **första undersökningen**, som var en utvärdering av det befintliga systemet, gjordes utan testpersoner. Vi valde att först själva göra en utvärdering av Systemet för att på så sätt vara mer förberedda inför den andra undersökningen där vi utvärderade Systemet tillsammans med användare.

Den första undersökningen är baserad på en systemutvärderingsmetod som Jacob Nielsen kallar för *Heuristisk Evaluering*²⁷. Enligt Nielsen är den Heuristiska Evalueringen billig att använda då den inte kräver några användare och då den går relativt fort att utföra. Metoden utgår från tio hållpunkter vilket gör att även personer som inte är vana att utvärdera datorsystem kan göra det på ett bra sätt (läs mer om den Heuristiska Evalueringen i kapitel 5.2.1).

I den **andra undersökningen** använde vi oss av etnografisk observation vilken beskrivs av Shneiderman²⁸. Enligt Shneiderman bygger den etnografiska observationen på att användargränssnittsdesigners observerar hur användaren använder systemet i syfte att ändra och förbättra användargränssnittet. Till skillnad från den traditionella etnografiska studien, där observatören är delaktig i organisationen i veckor eller månader, är detta en mer begränsad process vad gäller tiden. Det räcker med att observera ett fåtal dagar eller till och med några få timmar

²⁷ Nielsen, Jakob, *Usability engineering* (1993)

²⁸ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

för att få den relevanta informationen som behövs för att göra eventuella ändringar i systemet²⁹.

Genom att använda den etnografiska observationen när användarna använde tidsredovisningssystemet fick vi en bra inblick i användarnas problem och vilka ändringar som behövde göras. Vi observerade bl.a. hur mycket användarna letade i menyerna för att hitta information och funktioner, på vilket sätt användarna använde sig av funktionerna och hur de reagerade på olika händelser i Systemet.

För att ytterligare öka tillförlitligheten i denna undersökning valde vi att komplettera med en typ av intervju. Intervjun var egentligen mer en diskussion där en av oss förde ett samtal med användaren medan den andre observerade användaren under tiden som denne utförde uppgifter i systemet. Detta skedde vid ett och samma tillfälle. Huvudsyftet med diskussionen var att få användaren att prata högt medan hon eller han utförde uppgifterna. Frågorna som vi ställde var öppna och ostrukturerade, dvs vi hade inga förberedda frågor utan de växte fram beroende på situationen. Frågorna kunde var av typen: Varför gjorde du så där? Brukar du alltid göra så? Vad tänker du nu?

För att få ytterligare information om användarens uppfattning och attityd till tidsredovisningssystemet utformade vi, som ett komplement till observationen och intervjun, en enkät i form av ett attitydformulär som användaren fick svara på. Detta attitydformulär, som enbart bestod av strukturerade frågor med färdiga svarsalternativ, användes både vid utvärderingen av det befintliga systemet och vid utvärderingen av den utarbetade prototypen.

Vi avslutade undersökningen med en typ av semistrukturerad intervju som bestod av en fråga. Beroende på användarens svar ställdes ett antal följdfrågor. Syftet med intervjun var att få reda på vad användaren tyckte om Systemet och få förslag på eventuella förändringar.

I den **tredje undersökningen** utarbetade vi, i interaktion med användarna, en prototyp av användargränssnittet. Det etnografiska synsättet som låg till grund i denna undersökning var Hughes "quick and dirty ethnography"³⁰. Denna typ av etnografi går i vårt fall ut på att observatören gör sig bekant med användarens omgivning och dennes system samtidigt som en prototyp skapas av systemet. Till skillnad från den etnografiska observationen som beskrivs ovan grundar sig quick and dirty etnografi på att man tillsammans med användarna skapar en prototyp av ett systems användargränssnitt. Genom att göra upprepande genomgångar av prototypen, göra fler fältstudier och omdesigna prototypen ett antal gånger får man ett användargränssnitt som de flesta användare är nöjda med.

Genom att sammanföra informationen från de två tidigare undersökningarna fick vi en bra grund att utgå ifrån när vi designade den första prototypen av användargränssnittet. Vi träffade sammanlagt tre grupper, med två användare i varje grupp, som fick vara med och bestämma utformningen av användargränssnittet. Mellan varje träff med grupperna omdesignades användargränssnittet efter gruppens

²⁹ Ibid

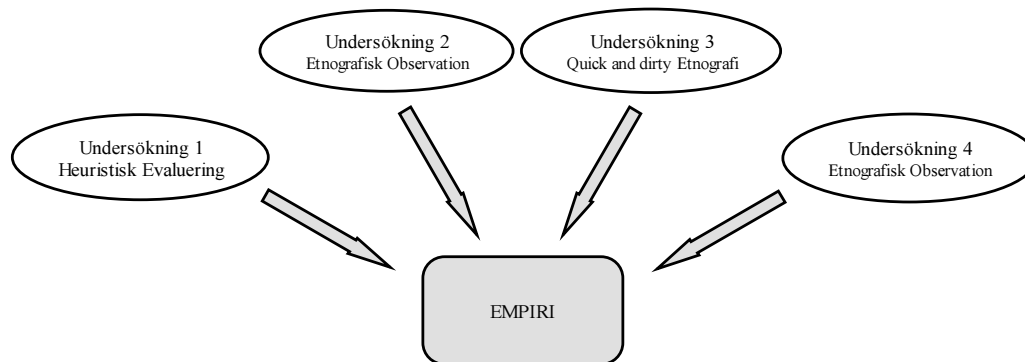
³⁰ Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design* (1994)

önskemål. Genom att på detta sätt gå fram och tillbaka mellan olika grupper av användare fick vi till slut en användargränssnittsprototyp som de flesta i stort var nöjda med.

Den **fjärde** och sista **undersökningen** vi gjorde gick ut på att utvärdera den framtagna användargränssnittsprototypen. Detta gjordes på ungefär samma sätt som vid utvärderingen av det befintliga systemet. I likhet med den andra undersökningen genomsyras denna undersökning av etnografisk observation. Vi observerade användaren när denne utförde uppgifter på prototypen. Användaren fick även fylla i ett attitydformulär.

Genom att vi gjorde observationer både när användarna använde det befintliga användargränssnittet och när de använde det nya, kunde vi direkt mäta om vi hade lyckats eller ej med designen av den nya användargränssnittsprototypen. Attitydformuläret hjälpte oss ytterligare i mätningen av användarnas uppfattning och attityd till det nya användargränssnittet.

Figur 2.5 nedan visar en sammanfattning över hur vi, genom våra undersökningar, gått tillväga för att samla den empiri som ligger till grund för denna studie.



Figur 2.5. De fyra undersökningarna som ligger till grund för vår studie

Del II Teori

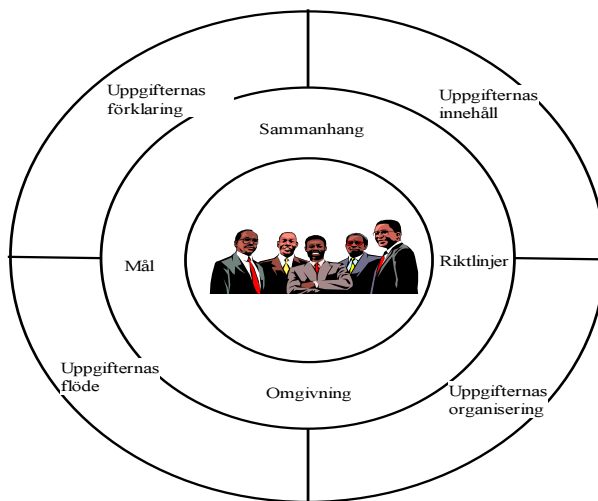
3. Användarcentrerad Design

Målet med ett datorsystem är att underlätta för människor i deras vardagliga arbete. Studier visar dock att datorsystem ofta är svåra att lära sig och att få användare behärskar all funktionalitet i ett system. Vad beror det på? Varför är det så att människor och datorer har svårt att fungera ihop³¹?

En av anledningarna är enligt Fossum³² att systemutvecklaren under mångåriga datorstudier har blivit formad till en "datorexpert" och har därför svårt att förutse vilka problem användaren kan tänkas få med ett system. För att få ett användbart system krävs det att de som arbetar med systemutveckling fokuserar mer på människan som skall använda systemet än på tekniken som ligger bakom det.

Användarcentrerad design, eller den engelska termen User-Centred-Design (UCD), är enligt Preece det mest fundamentala inom MDI³³. Ett annat vanligt begrepp, med samma betydelse, är "Usability Engineering" som används av bl.a. Nielsen³⁴.

Som nämnts tidigare fokuserar användarcentrerad design på människor, deras arbete och deras omgivning³⁵. Användarcentrerad design fokuserar inte bara på tekniken, processen, metoderna, och proceduren i designen av användbara system, utan även på filosofin att ha användaren i centrum under hela systemutvecklingsprocessen. Figuren nedan visar processen i användarcentrerad design. All systemutveckling sker med användaren i centrum. Ett systems mål, riktlinjer, innehåll och omgivning mm har fokus på användaren³⁶.



Figur 3.1. Användarcentrerad design³⁷

³¹ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

³² Ibid

³³ Preece, Jenny, *Human-computer-interaction* (1996)

³⁴ Nielsen, Jakob, *Usability engineering* (1993)

³⁵ Preece, Jenny, *Human-computer-interaction* (1996)

³⁶ Rubin, Jeffrey, *Handbook of usability testing* (1994)

³⁷ Ibid

En definition av användarcentrerad design är enligt Woodsons³⁸ ”the practice of designing products so that users can perform required use, operation, service, and supportive tasks with a minimum of stress and maximum of efficiency”.

Hix menar att man i användarcentrerad design måste fokusera på det som är bäst för användarna istället för vad som är snabbast och lättast för utvecklarna att implementera. En annan viktig faktor vad gäller användarcentrerad design är enligt Hix att man på ett tidigt stadium måste lära känna användarna för att på så sätt kunna designa ett system som passar dem³⁹.

Att involvera användaren i utvecklingen av ett system har blivit allmänt känt som en nyckel till att uppnå användbarhet i ett system. Användaren känner mindre motstånd och känner sig mer säker på det nya systemet om hon eller han känner sig delaktig i systemutvecklingen⁴⁰.

Användarcentrerad design är mer en systemutvecklingsfilosofi än en designtyp. Filosofin handlar om att designteamet har användaren som första prioritet, den handlar om att fokusera på användaren i varje del av utvecklingsprocessen. Användarcentrerad design kräver en mängd olika kunskaper, varav den kanske mest viktiga är kunskapen om slutanvändaren och dennes användning av systemet. Idag består ett systemutvecklingsteam av specialister som har kunskap inom områden som t ex systemutveckling, marknadsföring, användargränssnittsdesign, mänskliga faktorer och multimedia. Denna variation av expertis har idag blivit en norm för att bygga ett system med hög användbarhet⁴¹.

4. Användargränssnittsdesign

När man har förvässat sig om den nödvändiga funktionaliteten i systemet, säkerställt systemets pålitlighet, fastställt standarder för systemet samt satt upp en tidsplan och budget, kan man lägga fokus på design av användargränssnittet⁴².

Något av det första människor frågar efter när de diskuterar ett nytt system är användargränssnittet. Ett väl designat gränssnitt är mycket viktigt för användarna eftersom de genom detta ser vilka möjligheter systemet har. För de flesta användare är användargränssnittet systemet, då det är detta användaren ser⁴³.

Vilka uppgifter som systemet visar att det kan utföra och *hur* dessa presenteras kan vid många tillfällen vara av stor betydelse för en organisation och dess relation med sina kunder⁴⁴. Man måste vara medveten om att vad som anses vara en god design i en verksamhet kanske inte alls passar i en annan. Därför måste man förstå den verksamhet systemet skall verka i och vilka uppgifter användarna vill kunna utföra i systemet⁴⁵.

Det finns många olika möjligheter att förbättra ett system på. Har systemet röriga fönster, komplexa och tråkiga procedurer, otillfredsställande funktionalitet,

³⁸ Ibid

³⁹ Hix, Deborah & Hartson H.R., *Developing user interfaces: Ensuring usability through product and process* (1993)

⁴⁰ Ibid

⁴¹ Rubin, Jeffrey, *Handbook of usability testing* (1994)

⁴² Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁴³ Galitz, Wilbert, *The essential guide to user interface design* (1996)

⁴⁴ Ibid

⁴⁵ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction*, (1998)

inkonsekventa händelsesekvenser och otillräcklig informativ feedback kan användaren känna stress, förvirring och ångslan. Detta leder till att användaren får svårare att utföra sitt arbete och gör fler misstag⁴⁶.

I detta kapitel tar vi upp en mängd riktlinjer som man bör tänka på när man utformar ett användargränssnitt. Det är viktigt att följa dessa riktlinjer vid designen men man skall vara medveten om att detta inte är tillräckligt för att få en god design. Varje produkt och dess användare är unik och därför måste riktlinjerna skräddarsys för just den verksamhet i vilken systemet skall verka och för de unika krav som ställs på systemet⁴⁷. Detta tar tid och är en komplicerad process vilket är orsaken till att även om riktlinjerna kan tyckas vara självklara så följs de inte alltid vid design av användargränssnitt⁴⁸.

Något man alltid bör ha i bakhuvudet när man designar ett användargränssnitt är människors kognitiva förmåga. Med detta menas bl a att förstå, komma ihåg, vara medveten om, skaffa sig färdigheter och att skapa nya idéer⁴⁹. Vid design av användargränssnitt är man intresserad av användarnas förståelse av verkligheten samt deras olika färdigheter. Man bör designa ett gränssnitt på ett sätt som gör att det blir så enkelt som möjligt för användaren att utföra sina uppgifter i systemet. Ett system är till för att förenkla arbetsuppgifter och gränssnittet skall därför designas så att användaren i så liten omfattning som möjligt skall behöva använda sin intelligens för att klara av en uppgift i systemet⁵⁰.

En annan viktig aspekt att ta hänsyn till vid design av användargränssnitt är människors minnesbegränsningar och då främst kapaciteten och längden i människors sk korttidsminne. Kapaciteten mäts normalt efter Millers kända ”sju plus eller minus två enheter”⁵¹ där man med enhet menar den information som man måste komma ihåg. Uttrycket betyder att man endast kan komma ihåg mellan fem och nio ”informationsbitar” under en viss tid. Med minnets längd menar man längden av tiden som informationen kan hållas i minnet, vilket brukar vara mellan 30 sekunder och 2 minuter. Vid design av användargränssnitt bör man begränsa antalet informationsbitar som en användare måste ha i minnet under ett visst tillfälle. Man skall även se till att informationen i användargränssnittet är organiserad så att användaren inte behöver komma ihåg information från ett fönster till ett annat⁵².

Vi har valt att i denna uppsats ta upp de riktlinjer vi anser vara mest relevanta för vår studie, dvs endast de riktlinjer som är av intresse för vår undersökning. Detta är med andra ord endast en översikt och absolut ingen fullständig beskrivning av riktlinjer⁵³.

4.1 Riktlinjer för användargränssnittsdesign

Vid design av system talar man ofta om olika interaktionsstilar såsom fönster, menyer, formulär och dialogrutor genom vilka användarna använder sig av systemet. Förutom dessa interaktionsstilar tar Shneiderman även upp ett antal designområden som fokuserar på interaktionen mellan systemets funktioner och utseende. Han menar

⁴⁶ Ibid

⁴⁷ Mayhew, Deborah, *The usability engineering lifecycle* (1999)

⁴⁸ Hix, Deborah & Hartson H.R, *Developing user interfaces: Ensuring usability through product and process* (1993)

⁴⁹ Preece, Jenny, *Human-computer-interaction* (1996)

⁵⁰ Hix, Deborah & Hartson H.R, *Developing user interfaces: Ensuring usability through product and process* (1993)

⁵¹ Ibid

⁵² Ibid

⁵³ Ibid

att det är viktigt att ha en god balans mellan dessa båda faktorer. Exempel på dessa designområden är systemmeddelande, skärmdesign och färger.⁵⁴

Vid design av interaktionsstilar och designområden är målet att skapa ett användargränssnitt som är förnuftigt, begripligt, och bekvämt organiserat på ett sätt som är relevant för de uppgifter som användarna skall utföra i systemet⁵⁵.

Nedan beskriver vi de interaktionsstilar och designområden som är av intresse för vår studie och hur de skall utformas för att systemet skall vara så användbart som möjligt.

4.1.1 Fönster och Skärmdesign

All den interaktion som sker mellan en användare och ett system sker via ett fönster. För de flesta interaktiva system är skärmfönstret därför nyckelkomponenten för en lyckad design. Ett fönster som är rörigt och har för mycket information kan skapa irritation och ett inkonsekvent fönsterformat kan försvåra utförandet av en uppgift⁵⁶.

Det finns två typer av fönster; primära och sekundära fönster. Genom det *primära* fönstret genereras alla andra fönster i ett program och det är genom detta fönster som programmet stängs ner⁵⁷. Det primära fönstret är det första fönstret som visas på skärmen när man startar ett program och fönstret är nödvändigt för utförandet av varje funktion i systemet⁵⁸.

Ett *sekundärt* fönster genereras genom det primära fönstret och visas ovanpå det primära fönstret. Det skall kunna gå att ändra storlek och flytta ett sekundärt fönster samt skrolla i det. Många system använder sig av multipla sekundära fönster för att utföra en uppgift. Sekundära fönster används för att utföra underordnade, kompletterande eller stödjande uppgifter som är relaterade till objekt i det primära fönstret⁵⁹. Om många fönster är öppna samtidigt, är endast ett fönster aktivt⁶⁰.

Riktlinjer för fönsterdesign⁶¹:

- *Överdriv inte användandet av fönster*
Minimera antalet fönster som behövs för att utföra en uppgift. Fönster är ett medel för att utföra något. Blir det för många fönster kan detta förvirra mer än förenkla. Varje gång ett nytt fönster öppnas skall detta placeras på en fast position i användargränssnittet. Användaren kan därefter flytta fönstret om så önskas. Överlappande fönster är att föredra framför sidoordnade fönster.
- *Utseendet och beteendet hos det primära fönstret bör vara konsekvent*
Det primära fönstret bör vara en startpunkt för användaren att återvända till om denne tappar bort sig bland alla fönster. Fönstret bör då se ut som användaren lämnade det. Ett undantag är självklart då användaren har gjort något i det sekundära fönstret som skall förändra innehållet i det primära fönstret.
- *Använd olika fönster för olika, oberoende uppgifter*
Man skall alltid organisera fönster så att de stödjer de uppgifter som användaren skall utföra. Detta kan endast göras på ett lämpligt sätt genom en ordentlig och

⁵⁴ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁵⁵ Ibid

⁵⁶ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁵⁷ Hix, Deborah & Hartson H.R., *Developing user interfaces: Ensuring usability through product and process* (1993)

⁵⁸ Galitz, Wilbert, *The essential guide to user interface design* (1996)

⁵⁹ Ibid

⁶⁰ Hix, Deborah & Hartson H.R., *Developing user interfaces: Ensuring usability through product and process* (1993)

⁶¹ Ibid

klar analys av användarnas uppgifter. Människor tänker ofta i termer av de uppgifter de skall utföra, inte i funktioner eller program. Fönster måste därför organiseras för att stödja detta tänkande. Målet med designen är att stödja användarnas vanligaste arbetsuppgifter på det mest effektiva sättet.

- *Bibehåll skärmkonsekvens*
En god design ändras så lite som möjligt från ett fönster till en annat. Om objekt, såsom knappar, ord och ikoner, uppkommer på flera fönster, bör de finnas på exakt samma ställe på alla fönster. Denna konsekvens är viktig främst när det gäller plats, form och storlek på objekten.
- *Organisera fönstret för att hantera komplexiteten*
Ett sätt att förenkla en fönsterdesign är genom att eliminera onödig information. Det är viktigt att minimera tätheten mellan objekt på skärmen, särskilt vad gäller texten. Det är svårt för användaren att läsa ord som står för nära varandra. För tätt mellan objekten kan även göra att viktig information döljs för användaren som denne behöver för att utföra en uppgift i systemet.

En effektiv design måste erbjuda all nödvändig data i en sekvens som passar uppgiften som ska utföras. En meningsfull gruppering av objekt och en konsekvent sekvensering av dessa grupper stödjer utförandet av uppgifter i systemet. Grupperingar kan omges av blankt utrymme eller av en rektangel, men kan även separeras med hjälp av en skuggad bakgrund, färger eller ett annorlunda teckensnitt⁶².

4.1.2 Menyner

Oavsett vilket syfte ett system har måste det kunna tala om för användarna vilken information det besitter och vad det kan utföra. Detta görs genom att lista de val eller alternativ som användaren har till förfogande i systemet. Dessa listor kallas för menyer och kan visas på olika sätt. Exempel på menyer är ”pull-down”, ”pop-up”, ”radio-button” och ”push-button” menyer⁶³.

Menyer är effektiva eftersom användarna inte behöver komma ihåg olika kommandon utantill utan kan välja mellan olika alternativ. När menybegreppen är skrivna med familjära termer och är organiserade i en bekväm struktur och sekvens, är det lätt för användarna att hitta i menyerna. Det är särskilt effektivt för nya användare som är ovana vid terminologin. Det finns många riktlinjer att följa när man designar menyer för att förenkla användarnas arbete. Nedan följer några av dessa riktlinjer⁶⁴:

- *Organisera hierarkiska menyer utifrån de uppgifter användaren utför i systemet samt utifrån systemets funktioner*
Genom att skapa kategorier av liknande poster kan menyerna organiseras i hierarkiska trädstrukturer, vilket är nödvändigt när en samling poster växer. Det är viktigt att grupperingen på varje nivå är naturlig och begriplig för användaren så att denne lätt kan hitta i menyträdet. Om grupperingarna istället är obegripliga och om användarna är osäkra på vad de söker efter kan de lätt tappa bort sig i menyerna.

⁶² Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁶³ Galitz, Wilbert, *The essential guide to user interface design* (1996)

⁶⁴ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

En bred, grund trädstruktur är att föredra framför en djup, smal struktur. Man brukar säga att en meny bör bestå av fyra till åtta poster och helst inte mer än tre nivåer. Fler nivåer ökar risken för att användaren tappar bort sig⁶⁵.

- *Använd meningsfulla, uppgiftsorienterade grupperingar av menyval.*
Poster bör grupperas tillsammans i en meny på ett sätt så att de är begripliga för användarna och matchar strukturen i deras uppgifter. Detta kan ibland vara svårt på grund av att t ex kategorier överlappar varandra, poster inte passar in under någon speciell meny och att begreppen är svårbegripliga. För att lösa dessa problem finns föreslagna regler för hur man formar trädstrukturer:
 - Gruppera poster som är logiskt lika.
 - Skapa grupper som täcker alla möjligheter.
 - Se till att posterna inte överlappar varandra. Strukturera menyerna så att en undermeny endast associeras till en av de överliggande menyerna.
 - Använd terminologi som är familjär för användaren. Se dock till att posterna skiljer sig från varandra⁶⁶.
- *Ordna menyvalen meningsfullt*
Om inte menyposterna har en naturlig sekvens, så som t ex tids- eller nummerordning, kan designern ordna posterna efter exempelvis; alfabetisk ordning, gruppering av relaterade poster, de vanligast använda posterna först eller de viktigaste posterna först⁶⁷.
- *Använd en kortfattad beskrivning av menyvalen*
Nedanstående riktlinjer för hur man namnger posterna i menyerna kan anses självklara, ändå är det ofta som de inte följs⁶⁸.
 - Använd familjär och konsekvent terminologi.
 - Se till att posternas termer skiljer sig från varandra.
 - Använd konsekventa och koncisa fraser. Om betydelsen av ett val är självklart; använd endast ett ord för valet.
 - Börja varje val med ett nyckelord om det inte är möjligt att använda enskilda ord. Det är exempelvis bättre att skriva "Size of type" istället för "Set the type size"
- *Var konsekvent i layouten av menyerna och se till att inte skärmen blir rörig*
Genom att vara konsekvent i menyformaten bidrar man till att användarnas förväntningar på vad som finns under de olika menyerna uppfylls, vilket i sin tur minskar deras osäkerhet. Konsekvens bör särskilt fastställas för följande komponenter⁶⁹:
 - Titlar. Exempelvis centrerad eller vänsterjusterad.
 - Placering av poster. Oftast vänsterjusterat. Man kan använda blanka rader för att separera grupper av poster i menyerna.
 - Instruktioner. Dessa begrepp bör vara konsekventa och bör visas på samma plats för varje meny.

⁶⁵ Ibid

⁶⁶ Ibid

⁶⁷ Hix, Deborah & Hartson H.R, *Developing user interfaces: Ensuring usability through product and process* (1993)

⁶⁸ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁶⁹ Ibid

4.1.3 Formulär

Ett formulär är ett fönster som består av namngivna fält som kan fyllas i av en användare, dels genom att skriva i dem och dels genom menyval. Formulär är populära tack vare att informationen är synlig vilket gör att användaren får en känsla av att ha kontroll över dialogen.

Riktlinjer för design av formulär⁷⁰:

- *Använd meningsfulla rubriker och begripliga instruktioner.*
Skriv förklaringar med termer som är familjära för användaren. Var kortfattad och var konsekvent vad gäller den grammatiska stilen i instruktionerna.
- *Använd en logisk gruppering och ordningsföljd på fälten.*
Relaterade fält bör placeras nära varandra och grupperingar bör separeras med blankt utrymme. Placera fälten konsekvent i alla formulär.
- *Skriv fältens rubrik med familjär och konsekvent terminologi.*
Varje fält bör alltid ha en rubrik som talar om vad användaren skall göra i fältet. Denna rubrik bör vara beskriven med en terminologi som stämmer med användarens begreppsvärld. Vanligast är att den är placerad till vänster om fältet som den tillhör. Varje gång ett fält visas i ett annat formulär bör fältet ha samma rubrik och, i största möjliga mån, vara på samma ställe.
- *Använd konsekventa och informativa felmeddelanden för oacceptabla tecken.* Om användare skriver in ett oacceptabelt värde, bör felmeddelandet visas omedelbart.

4.1.4 Systemmeddelanden⁷¹

Ett av de enklaste och mest effektiva sätten att förbättra ett existerande system är genom att förbättra felmeddelandena. Hur felmeddelanden och varningar är formulerade är av stor betydelse. Felmeddelanden kan vara den del av ett interaktivt system som har störst psykologisk påverkan på användare. Om ett meddelande har en fördömande ton, kan detta öka användarens ångslan, vilket kan bidra till att användaren får det svårare att rätta till felet.

Riktlinjer för att skriva systemmeddelanden:

- *Använd specifika termer*
Om ett meddelande är alltför generellt, finns risk att användaren inte förstår vad som är fel. Enkla meddelanden som även har en dömande ton kan skapa frustration då de varken ger tillräckligt med information om vad som är fel eller information som hjälper användaren att rätta till felet. Exempelvis är det bättre att skriva ”Dagarna sträcker sig mellan 1 och 31” istället för ”INVALID DATA”.
- *Ge konstruktiv vägledning och använd en positiv ton.*
Meddelanden bör i största möjliga mån indikera vad användarna behöver göra för att rätta till felet istället för att fördöma användarna för felet de har gjort. Negativa ord som ”ILLEGAL”, ”ERROR” och ”INVALID” bör antingen tas bort helt eller användas sällan då de kan göra att användaren känner sig hotad och osäker.
- *Använd användarcentrerade meddelanden*
Man bör alltid kommunicera med användare med termer utifrån deras begreppsvärld och utifrån deras arbetsuppgifter. Meddelanden som är centrerade kring systemet bör undvikas helt då dessa oftast förvirrar och gör användaren

⁷⁰Hix, Deborah & Hartson H.R, *Developing user interfaces: Ensuring usability through product and process* (1993)

⁷¹ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

osäker. Ett exempel på ett systemcentrerat meddelande kan vara ”505 hex 0001F9 doublewords of storage were not recovered” som visades vid användning av ett vanligt ordbehandlingsprogram. Denna information förstås enbart av den som är systemansvarig och är inte av intresse för någon annan. Denna typ av meddelande kan lätt skapa oro och panik hos en vanlig användare.

- *Använd ett lämpligt fysiskt format*
Det är lättare att läsa text som har en blandning mellan stora och små bokstäver. Meddelanden med endast versaler bör endast användas för korta, allvarliga varningar. Ett meddelande bör inte börja med ett långt, mystiskt kodnummer. Om det måste finnas med, bör det stå inom parenteser i slutet av meddelandet.
- *Var konsekvent*
Systemmeddelanden bör visas på samma ställe i användargränssnittet och bör ha en konsekvent terminologi och syntax.

4.1.5 Färger

Färgade användargränssnitt kan förbättra utförandet av en uppgift och göra användargränssnittet mer attraktivt för användaren, samtidigt som vissa färger och färgkombinationer kan vara negativt för användaren. Färger kan både lugna och irritera ögat, de kan göra så att användaren fokuserar på ett ointressant fönster eller del av ett fönster, skapa en logisk organisation av information, dra uppmärksamhet till varningar och skapa känslomässiga reaktioner av glädje, spänning, rädsla och ilska⁷².

Riktlinjer för användning av färger⁷³:

- *Begränsa antalet färger.* En generell regel är att inte använda mer än fyra olika färger i ett fönster och inte mer än sju färger i hela användargränssnittet.
- *Använd färger för att visa relationer.* Färger kan visa relationer mellan objekt på datorskrämen, antingen mellan relaterade fält i ett enskilt fönster eller mellan relaterade fält i olika fönster.
- *Använd färger konservativt.* Alltför många färger kan pigga upp ett fönster men kan också göra det svårläst. Istället för att skapa meningsfulla relationer, kan opassande färgsättning av fält vilseleda användaren till att söka efter relationer som inte finns.
- *Var konsekvent vid färgkodning.* Använd samma regler vid kodning av färger genom hela systemet. Om vissa felmeddelanden är röda bör alla felmeddelanden vara röda.
- *Var uppmärksam på färgers betydelse.* Färger har olika betydelse för olika människor. Designern måste vara medveten om och följa betydelsen av de färger som används i den domän där systemet skall verka.
- *Var uppmärksam på problem med färgkombinationer.* En generell regel är att blått och svart gör sig bäst som bakgrundsfärg, särskilt för vit respektive gul text. Blått bör inte användas för text då det är en av de svåraste färgerna att läsa⁷⁴. Blått och rött är en dålig kombination. Om de visas på ett fönster samtidigt kan det vara svårt för användaren att ta till sig informationen.
- *Ta hänsyn till färgblindhet.* En del av befolkningen är färgblind varför man bör se till att gränssnittet kan användas utan färger. Man bör därför inte visa viktiga relationer mellan olika objekt med endast färger⁷⁵.

⁷² Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

⁷³ Ibid

⁷⁴ Hix, Deborah & Hartson H.R., *Developing user interfaces: Ensuring usability through product and process* (1993)

⁷⁵ Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction* (1998)

5. Utvärderingsmetoder

Som tidigare nämnts i uppsatsen är användbarhet en viktig faktor när man utvecklar datorsystem. För att man skall kunna bygga ett användbart system är det viktigt att man i tidigt skede testat systemet på slutanvändarna. Dessa tester går ut på att hitta de användbarhetsproblem som användaren upplever i systemet⁷⁶.

Syftet med att leta efter användbarhetsproblemen är att försöka åtgärda dem innan systemet tas i bruk. Om systemet har brister i användbarheten får användarna spendera mycket onödigt tid på inläring och felkorrigering. Ett företag som använder sig av ett system med låg användbarhet drabbas i form av frustration hos användarna och ineffektivitet i arbetet. Genom att använda sig av utvärderingsmetoder i systemutvecklingen kan man identifiera och åtgärda många av de användbarhetsproblem som användaren upplever med systemet. Om man sedan upprepar användningen av utvärderingsmetoderna under systemutvecklingsprocessen får man ett bra mått på om användbarhetsproblemen har minskat eller ej. Ju tidigare man hittar problemen desto lägre blir kostnaderna för att rätta till dem⁷⁷. Pressman⁷⁸ menar att det är 60-100 gånger dyrare att genomföra systemförändringar efter att systemet tagits i bruk än att genomföra samma ändringar under systemets tidigare systemutvecklingsfaser.

Men hur går man till väga för att utvärdera ett datorsystem? Tyvärr finns det inte några gyllene regler för hur man skall gå tillväga. Varje system är unikt och är designat för olika användare som i sin tur har olika arbetsuppgifter. Vidare skiljer sig användarna från varandra, exempelvis så har förstagångsanvändare andra krav i jämförelse med de som har använt systemet en längre tid⁷⁹.

För att lyckas med utvärderingen är det nödvändigt att använda mer än en utvärderingsmetod under systemutvecklingsprocessen. Preece⁸⁰ rekommenderar att man skall använda en lätt utvärderingsmetod som går snabbt att utföra tidigt i systemutvecklingen, medan de mer omfattande utvärderingsmetoderna bör användas senare i utvecklingsarbetet. Vilken typ av metod som skall användas beror på var i systemutvecklingen man befinner sig, vilken typ av system man bygger, vilka som skall använda systemet och vilka ekonomiska resurser som finns till förfogande.

Figuren nedan visar att utvärderingen, enligt Preece, är den mest centrala i systemutvecklingsprocessen. Utvärderingen hjälper till i designprocessen genom att informera designteamet om hur väl designen passar användarna.

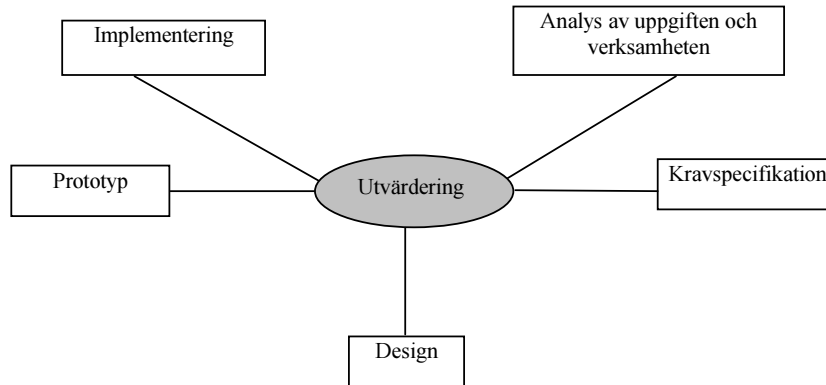
⁷⁶ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

⁷⁷ Ibid

⁷⁸ Pressman, R.S. *Software Engineering: a practitioner's approach* (1994)

⁷⁹ Preece, Jenny, *Human-computer-interaction* (1996)

⁸⁰ Ibid



Figur 5.1. "The star life cycle"⁸¹

Enligt Rubin⁸² är målen med att testa ett system följande:

- Skapa historik i användbarhet för framtida realiseringar så att man inte gör samma misstag igen.
- Minimera kostnaden i form av service och support.
- Öka försäljningen och möjligheten till återköp.
- Skapa en konkurrensfördel, då användbarhet har blivit ett sätt att skaffa sig marknadsandelar.
- Minimera risken att realisera en produkt med allvarliga användbarhetsproblem.

5.1 Val av utvärderingsmetod

Val av utvärderingsmetod beror på olika faktorer. En av faktorerna är *var i systemutvecklingen man befinner sig*. Vissa metoder är bättre lämpade än andra att använda i början av systemutvecklingen då de inte kräver att man har ett färdigt system. Det kan då räcka med att systemet finns skisserat på papper⁸³.

En annan faktor som man bör ta hänsyn till när man väljer utvärderingsmetod är vilken *typ av problem man vill hitta*. Genom att använda sig av utvärderingsmetoder som inkluderar användare upptäcks de problem som användaren upplever med systemet. Med metoder som inte inkluderar användare hittar man aspekter i systemet som användaren troligtvis kommer att uppfatta som användbarhetsproblem⁸⁴.

Val av metod är också *en ekonomisk- och tidsrelaterad fråga*. Ju mer resurser i form av tid och pengar man har till förfogande, ju fler och dyrare utvärderingsmetoder kan man utföra. Den största skillnaden, vad gäller tid och pengar, är mellan de metoder som inkluderar användare och de som inte gör det⁸⁵.

Om det är *svårt att få tag på användare* är val av metod begränsad. Om man vill utföra tester på systemet men inte har tillgång till slutanvändarna kan man använda sig av "låtsasanvändare". Dessa kan exempelvis vara systemutvecklare eller andra

⁸¹ Hix, Deborah & Hartson H.R., *Developing user interfaces: Ensuring usability through product and process* (1993)

⁸² Rubin, Jeffrey, *Handbook of usability testing* (1994)

⁸³ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

⁸⁴ Ibid

⁸⁵ Ibid

personer som finns i ens omgivning som kan ställa upp och agera användare av systemet. Har man inte heller denna möjlighet får man använda sig av de utvärderingsmetoder som inte inkluderar användare⁸⁶.

En annan aspekt vid val av utvärderingsmetod är *vilken kompetens som finns att tillgå*. Personer som har MDI-kunskap och/eller erfarenhet av utvärderingsmetoder kan utföra metoderna mer professionellt, vilket innebär ett bättre resultat.

Det finns en uppsjö av metoder för att hitta användbarhetsproblem hos datorsystem. Det finns till exempel en metod där användaren sitter i ett rum och blir observerad genom ett sk ”one-way-window” när hon/han arbetar med systemet. Vidare finns det en metod, sk ”loggingmetod”, som räknar antal tangenttryckningar och musklick en användare gör när denne använder systemet. Detta görs med ett speciellt datorprogram. En annan metod, vilken är ganska tidskrävande, är ”Retrospektiv testning”. Denna metod går ut på att användaren videofilmas då hon/han arbetar med systemet. Användaren och testledaren tittar därefter tillsammans på videon och användaren berättar hur hon/han upplevde de olika delarna i systemet⁸⁷.

Metoderna ovan kräver vissa resurser som till exempel laboratorium, loggingprogram och videokamera. Vi kommer inte att beskriva dessa utvärderingsmetoder närmare i denna uppsats, utan kommer endast att beskriva de metoder som är relevanta för vår undersökning. De ovannämnda metoderna har vi valt bort, dels på grund av att de kräver speciella resurser som vi inte har tillgång till och dels då vi anser att de metoder vi valt att använda i denna studie ger oss ett tillförlitligt resultat.

I litteraturen förekommer det att en och samma utvärderingsmetod har olika namn, vilket kan vara något förvirrande. Vi har valt att följa Jacob Nielsens⁸⁸ förslag på metodnamn. Nielsen delar in utvärderingsmetoderna i tre grupper, Inspektion, Användartest och Empirisk Prövning⁸⁹. Förutom dessa tre typerna av utvärderingsmetoder har vi lagt till en fjärde typ, som kallas Prototyping. Vi har valt att göra denna indelning för att öka förståelsen i denna ”djungel” av utvärderingsmetoder.

⁸⁶ Ibid

⁸⁷ www.cs.umd.edu/~zzj/UsabilityHome.html

⁸⁸ Nielsen, Jakob, *Usability engineering* (1993)

⁸⁹ Nielsen, Jakob, *Usability engineering* (1993)

Nedan har vi gjort en sammanfattning av de utvärderingsmetoder som vi kommer att ta upp i denna uppsats. Vidare i detta avsnitt görs en mer ingående beskrivning av dessa metoder.

Metod namn	Typ av metod	Antal användare	Fördelar	Nackdelar
Heuristik Evaluerings	Inspektion	Inga	Tekniken är baserad på de tio tumreglerna som är relativt lätt att följa för utvärderaren. Metoden är snabb, billig och lätt att genomföra.	Metoden involverar inga användare. Det är svårt för utvärderarna att hitta alla användbarhetsproblem då de inte har kunskap om slutanvändarna och deras arbetsuppgifter.
Standard Inspektion	Inspektion	Inga	Ett prototypverktyg har utarbetats för att underlätta i inspektionsprocessen för att automatiskt kontrollera om systemet skiljer sig från en viss standard. Det underlättar betydligt om ett system följer en speciell standard. Det gör att systemet blir lättare att förstå och lära sig.	Det är en tidskrävande metod och det är inte säkert att man hittar alla de delar av systemet som avviker från den valda standarden. Metoden fokuserar bara på avvikelserna och tittar inte på andra användbarhetsproblem i systemet.
Tänka-högt- protokoll	Användartest	3-5 st	En lätt och smidig metod för att få förståelse för hur användaren upplever systemet. Testet är relativt billigt att utföra.	Onaturligt för användaren att hela tiden "tänka högt". Vana användare kan ha svårt att tänka högt när de använder systemet, då de vanligtvis inte funderar så mycket över hur systemet fungerar.
Samupptäckar- metoden	Användartest	6 st	Denna metod kan kännas mer naturlig för användarna då de är vana att diskutera med andra när de löser problem. Mycket användbara kommentarer kommer från användarna.	Denna metod är dyrare än t ex Tänka-högt-protokollet p.g.a. av att man behöver fler användare. Vidare är det svårare att observera två personer vid samma tillfälle.
Coaching Metoden	Användartest	4 st	En bra metod att använda om det är svårt att få tag på användare p.g.a. att användarpopulationen är liten eller högavlönad. Vidare lär sig användaren systemet ganska snabbt.	Tekniken kräver en expert av systemet. Det kan vara svårt att hitta en expert som har tid att vara med i användartestet.
Fältobservation	Empirisk Prövning	2 st	Den enklaste av alla utvärderingsmetoder. Metoden ger information om användarens omgivning och hur de använder systemet.	Det kan kännas obekvämt för användaren att bli observerade när de utför sitt arbete. Det kan också vara svårt för observatören att förstå varför användaren gör saker och ting på ett speciellt sätt.
Fokusgrupp	Empirisk Prövning	6-9 per grupp	Denna metod kan fånga användarnas spontana reaktioner och idéer vad gäller systemet.	Informationen man får från fokusgruppen tenderar att ha en låg validitet. Det är även svårt att analysera informationen. Då man behöver åtminstone två fokusgrupper, dvs 12-18 personer, för att få ett representativt resultat, kan det vara svårt att få tag på användare.
Intervju	Empirisk Prövning	5 st	Användbar metod för att få reda på vad användarna behöver och vad de gillar och inte gillar i systemet. Svarsfrekvensen är hög och det finns möjlighet att upprepa eller förtydliga frågan om så skulle behövas.	Tidskrävande metod. Det kan vara svårt att analysera och jämföra svaren. Användarna kan ibland säga saker som de tror att intervjuaren vill höra. Användarna vill inte säga negativa saker om systemet p.g.a. att de tror att intervjuaren är en i systemutvecklingsteamet.
Enkät	Empirisk Prövning	Minst 30	Den enda metoden som kan upptäcka skillnader mellan olika användarkategorier och speciella krav från olika grupper av användare. Andra fördelar med metoden är att det går att använda samma frågor vid ett senare tillfälle för att kontrollera om användarna har ändrat uppfattning.	Man måste vara enormt försiktig när man väljer vilka frågor som skall vara med i enkäten. Det kan därför ta ganska lång tid att utforma enkäten. Det kan också ta lång tid att få tillbaka enkäterna och man får alltid räkna med ett antal bortfall.
Lo-fidelity prototyp	Prototyping	3-6	Metoden kan användas när som helst i systemutvecklingsprocessen. Det går snabbt och enkelt att utföra metoden.	Kan vara svårt för användarna att tänka sig hur användargränssnittet kommer att se ut då man endast visar det i pappersform.

Figur 5.2. Sammanställning av utvärderingsmetoder

5.2 Inspektion

Inspektionsmetoden innebär att man inspekterar användbarhetsaspekter i användargränssnittet och den utförs för det mesta utan användare⁹⁰. Det är framför allt specialister inom MDI och/eller personer med liknande kunskap som gör inspektionen⁹¹.

Två av de mest vanliga Inspektionsmetoderna är:

- Heuristisk Evaluering
- Standard Inspektion

5.2.1 Heuristisk Evaluering

Heuristisk Evaluering, som är framtagen av Jakob Nielsen och Rolf Molich, är en metod där utvärderare inspekterar ett system eller en prototyp med hjälp av en enkel och generell checklista med tio tumregler⁹². Denna metod är relativt billig och enkel att utföra, dels då den inte inkluderar användare och dels för att utvärderingen stödjer sig på tio tumregler. Detta gör att metoden lämpar sig för företag som kanske inte har ekonomiska resurser, tid eller expertis att använda metoder som inkluderar användare⁹³.

Syftet med metoden är att identifiera avvikelser från tumreglerna hos det granskade systemet. Avvikelserna är de användbarhetsproblem som systemet besitter och bör därför rättas till i nästa version av systemet⁹⁴.

Den Heuristiska Evalueringen grundar sig på att ett flertal utvärderare, oberoende av varandra, utvärderar ett system för att komma fram till eventuella användbarhetsproblem. Det är inte tillåtet för utvärderarna att diskutera sinsemellan under själva utvärderingen. Utvärderarna går normalt två eller fler gånger igenom systemets användargränssnitt. De börjar med att inspektera flödet av gränssnitten från fönster till fönster och därefter inspekterar de ett fönster i taget med hjälp av de tio tumreglerna⁹⁵.

Efter att alla utvärderare har avslutat sin individuella inspektion är det tillåtet att diskutera med de andra inspektörerna för att jämföra resultaten⁹⁶.

Figuren nedan visar att man med hjälp av t ex cirka fem utvärderare kan upptäcka ca 75 procent av alla användbarhetsproblem⁹⁷. Dessa problem kan dock vara annorlunda i jämförelse med de problem som man hittar när man utför tester med användare och på så sätt får deras perspektiv. Därför kan det vara bra att komplettera den Heuristiska Evalueringen med en eller fler utvärderingsmetoder som inkluderar användare⁹⁸.

⁹⁰ Preece, Jenny, *Human-computer-interaction* (1996)

⁹¹ Nielsen, Jakob, *Usability engineering* (1993)

⁹² Ibid

⁹³ Preece, Jenny, *Human-computer-interaction* (1996)

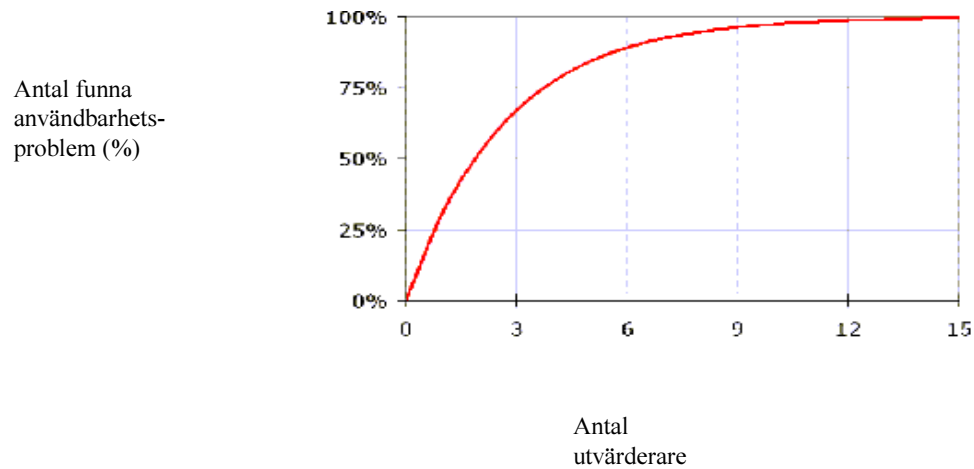
⁹⁴ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

⁹⁵ Preece, Jenny, *Human-computer-interaction* (1996)

⁹⁶ Nielsen, Jakob, *Usability engineering* (1993)

⁹⁷ Ibid

⁹⁸ Preece, Jenny, *Human-computer-interaction* (1996)



Figur 5.3. Relationen mellan antal utvärderare och antal funna användbarhetsproblem

Fördelen med den Heuristiska Evalueringen är att den är lätt att lära sig och den går relativt fort att genomföra. En annan fördel med metoden är att den kan användas både på ett färdigt användargränssnitt och på skisser av det blivande systemet, vilket gör att metoden är särskilt lämplig att använda i de tidiga faserna av systemutvecklingen⁹⁹.

⁹⁹ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

Figuren nedan beskriver de tio tumreglerna enligt den Heuristiska Evalueringen

1. **Använd en enkel och naturlig dialog:** För att underlätta för användaren bör man endast presentera information som användaren behöver för stunden. Vidare bör systemets användargränssnitt stödja användarens naturliga sätt att lösa uppgifterna. Färger och den grafiska designen skall på största möjliga sätt förenkla tolkningen av informationen.
2. **Tala slutanvändarens språk:** Användargränssnittet skall i största möjligast mån vara på användarens modersmål. Detta gäller bl.a. menyer, knappar, dialoger och val av ikoner. Val av termer och begrepp i användargränssnittet skall basera sig på användarnas begreppsvärld.
3. **Minimera användarens minnesbelastning:** Genom att vara konsekvent och minimera de detaljer och begrepp som användaren måste lära sig utantill, minimerar man användarens minnesbelastning. Detta bidrar även till att systemet blir lätt att lära sig.
4. **Var konsekvent:** Om användarna vet att samma kommandon och samma handlingar alltid gör samma saker känner de sig säkrare och vågar därmed prova andra funktioner i systemet. Man bör även ha konsekventa namn på termer och begrepp i systemet som stödjer det mänskliga igenkännandet.
5. **Ge återkoppling (feedback):** Systemet skall hela tiden informera användaren vad det gör och hur det tolkar användarens inmatningar. Återkoppling bör ges både när systemet utfört ett kommando korrekt och när det misslyckats.
6. **Ha tydliga vägar tillbaka:** Systemet bör alltid kunna erbjuda vägar tillbaka (t ex avbryt) från dialoger och andra komponenter. Det är även viktigt att användaren kan ångra sig i systemet. På så sätt vågar användaren utforska systemet i större utsträckning.
7. **Stöd genvägar:** Ett system bör vara uppbyggt så att både nya och mer erfarna användare kan utföra sina arbetssysslor på ett effektivt sätt. Kortkommandon och ikoner är ett sätt att stödja detta på.
8. **Ge bra felmeddelanden:** Det är viktigt att systemet ger tydliga och förståeliga felmeddelanden. Felmeddelandet skall vara välformulerat och ge tillräcklig information för att hjälpa användaren att lösa problemet. Meddelandet får vidare inte lägga skulden på användarna, då det kan bidra till att användarna inte vågar försöka igen.
9. **Förhindra fel:** Om det går att göra fel i system kommer användaren någon gång att göra det. Genom att utforma system som förhindrar att fel uppstår känner sig användaren säkrare vid användningen av det.
10. **Ge bra hjälp och dokumentation:** Även om de flesta användare inte läser manualer eller hjälpdialoger bör den som är intresserad kunna läsa och förstå dessa snabbt och effektivt. Det krävs då att informationen är strukturerad och pedagogiskt skriven.

Figur 5.5. De tio tumreglerna för att designa ett användbart användargränssnitt

Varje person som inspekterar systemet kommer att hitta ett antal punkter som inte stämmer överrens med de tio tumreglerna. Enligt Nielsen utgör dessa avvikelser systemets användbarhetsproblem och bör därmed åtgärdas.

Resultatet av den Heuristiska Evalueringen är en lista med systemets användbarhetsproblem. Alla listor från utvärderarna sammanställs till en lista som ges till systemutvecklarna för att de i nästa systemdesign skall kunna åtgärda problemen. Det är viktigt att utvärderarna har preciserat varje problem för sig, dels då det på det viset blir lättare att prioritera bland problemen, och dels för att systemutvecklarna lättare skall förstå problemen. För att utvärderarna och systemutvecklarna verkligen skall vara säkra på att de förstår varandra bör de ha ett gemensamt genomgångsmöte. Syftet med mötet är att utvärderarna kan förklara användbarhetsproblemen för systemutvecklarna och på så sätt tillsammans diskutera och lösa de problem som finns i systemet¹⁰⁰.

¹⁰⁰ Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem* (1996)

5.2.2 Standardinspektion

Det finns en mängd olika standarder för datorsystem med grafiska användargränssnitt, t ex Microsoft har standarder för Windows, Macintosh har standarder för Apple och Motif har sina standarder för Unix. Vissa företag har även en egen standard för sina system.

Det är inte valet av standarden som är avgörande för om ett datorsystem är lätt att använda eller inte. Det är viktigare att vara konsekvent, dvs att man följer samma standard genom hela systemet. Det är enklare för användaren om alla program som hon eller han jobbar med följer samma standard. Det är på det sättet enklare för användaren att lära sig och förstå nya program.

Standardinspektionen kontrollerar om systemet kan orsaka användbarhetsproblem genom att det skiljer sig från en speciell standard. Till exempel förväntar sig användare som är vana vid att använda Windowsprogram att andra program har samma genvägar och samma alternativ i menyerna som Windowsprogrammen¹⁰¹.

Metoden kräver att man har expertis på standarder. Experterna går igenom användargränssnittet eller prototypen för att få en känsla för systemets omfattning. Därefter hålls ett möte där experterna och representanter för utvecklingsteamet är närvarande. På mötet går man noggrant igenom användargränssnittet och experterna får peka ut var i systemet som det avviker från standarden. Det är sedan upp till systemutvecklarna att åtgärda dessa avvikelser¹⁰².

Standardinspektionen är viktig då systemet blir lättare för användaren att använda om systemet konsekvent följer en standard som användaren är van att använda. En nackdel med metoden är att den är tidskrävande. Vidare så fokuserar Standardinspektion bara på avvikelser från standarden och inte på andra användbarhetsproblem i systemet. Därför kan det vara bra att komplettera med en annan metod för att hitta andra typer av problem i systemet¹⁰³.

5.3 Användartest

Användartest, eller användbarhetstest, bygger på att slutanvändare gör typiska uppgifter i systemet eller på prototypen. Resultatet från ett användartest visar i hur hög grad användargränssnittet hjälper användaren att förstå hur hon eller han skall utföra uppgifter i systemet¹⁰⁴.

De vanligaste användartestmetoderna och de metoder vi anser vara av intresse för vår studie är följande:

- Tänka-högt-protokoll (Think Aloud Protocol)
- Samupptäckar-metoden (Co-discovery Learning)
- Coaching Metoden (Coaching Method)

¹⁰¹ Ibid

¹⁰² Nielsen, J & Mack, R.L, *Usability Inspection Methods* (1994)

¹⁰³ Ibid

¹⁰⁴ Nielsen, Jakob, *Usability engineering* (1993)

5.3.1 Tänka-högt-protokoll

Detta användartest bygger på att den framtida användaren utför uppgifter i systemet medan hon eller han "tänker högt". Genom att låta användarna prata högt när de använder systemet får testledaren en förståelse för hur användaren upplever det och kan därmed lätt identifiera de största användbarhetsproblemen.

För att hjälpa testpersonen att tänka högt kan testledaren ställa frågor som exempelvis "vad tänker du nu?" och "vad tror du menas med detta meddelande?" Testledaren bör inte svara på frågor som till exempel "Kan jag göra så här?" men kan däremot uppmuntra testpersonen att prata vidare genom att ställa en motfråga som "vad tror du händer om du gör det?"¹⁰⁵.

En fördel med metoden är att man kan samla information från endast ett fåtal användare. Användarnas kommentarer om hur de upplever användargränssnittet är viktig information för att skapa ett användbart system¹⁰⁶.

En nackdel är att det kan kännas onaturligt för många människor att "tänka högt". Kravet att hela tiden prata kan göra att användarens vanliga arbetstakt går långsammare, vilket kan innebära att användarens vanliga beteende vid problemlösning påverkas¹⁰⁷.

5.3.2 Samupptäckar-metoden

Detta är en variant av Tänka-högt-protokoll, men i denna metod är det två användare som tillsammans använder systemet för att lösa uppgifter. Till skillnad från Tänka-högt-protokoll, är denna metod mer naturlig för användarna. Detta då människor är vana vid att kommunicera med varandra då de löser problem vilket gör att de på ett naturligt sätt "tänker högt".

Metoden går ut på att två användare får utföra uppgifter i systemet samtidigt som de uppmuntras att kommunicera med varandra när de arbetar med systemets användargränssnitt¹⁰⁸. Rubin¹⁰⁹ menar att dialogen mellan användarna är så pass viktig att den förmodligen behövs bandas och skrivas ner. Det är också viktigt att testledaren hela tiden uppmuntrar användarna att "tänka högt", även i de fall då det inte behövs mycket kommunikation för att lösa en uppgift¹¹⁰.

Nackdelen med metoden är att olika användare har olika strategier för att använda en dator och dess system, vilket kan resultera i att två användare kan ha svårt att samarbeta. Ett sätt att lösa detta dilemma är att låta två användare som känner varandra sedan tidigare arbeta tillsammans¹¹¹.

5.3.3 Coaching Metoden

Den sista användartestmetoden som vi valt att ta upp är Coaching Metoden. Denna metod skiljer sig lite från de andra användartesterna genom att den tillåter användaren att ställa frågor om systemet. Frågorna besvaras av en person som är expert på

¹⁰⁵ Nielsen, Jakob, *Usability engineering* (1993)

¹⁰⁶ Ibid

¹⁰⁷ Ibid

¹⁰⁸ Ibid

¹⁰⁹ Rubin, Jeffrey, *Handbook of usability testing* (1994)

¹¹⁰ Ibid

¹¹¹ Nielsen, Jakob, *Usability engineering* (1993)

systemet. Experten är med under hela testet och styr användaren i rätt riktning när denne använder systemet. Coaching Metoden är mer naturlig än Tänka-högt protokollet då användaren hela tiden kan fråga experten om det är något som denne inte förstår i systemet. Ett syfte med metoden är att ge användaren bättre träning i att använda systemet. Ett annat syfte är att upptäcka vilken information som användaren behöver för att på så sätt skapa en bra dokumentation. Informationen från användarna ligger till sedan grund för omdesignen av användargränssnittet¹¹².

5.4 Empirisk Prövning

Användartester, som vi beskrivit ovan, är hörnstenen bland utvärderingsmetoderna, men det finns andra metoder som bör användas för att komplettera informationen om användarens uppfattning och attityd till systemet¹¹³. Vid empirisk prövning fokuserar man på användarna som får svara på frågor verbalt eller skriftligt vad de tycker och vad de behöver i systemet. Utvärderaren kan också få information om användarna genom att observera dem när de använder systemet i sitt dagliga arbete. Empirisk Prövning inkluderar:

- Fältobservation
- Fokusgrupper
- Intervju
- Enkät

5.4.1 Fältobservation

Fältobservation går ut på att observera användarnas omgivning och deras arbete för att på så sätt förstå hur de använder systemet i sitt dagliga arbete. Det är bra att välja en blandning av representativa användare från olika arbetsplatser, branscher och bakgrund.

Inför observationen bör observatören förbereda en lista med vad som skall observeras, dvs vilken information som observatören vill samla in. Under observationen skall observatören vara så tyst som möjligt och i största möjligaste mån inte lägga sig i användarnas arbete utan istället noterar de händelser som hon/han inte förstår under observationen. När observationen är klar kan observatören fråga användaren om det som varit oklart. Nielsen menar att observatören i slutet av en observation kan gå ur sin roll som observatör och istället hjälpa användaren med de uppgifter som denne hade problem med i systemet. Observatören kan även svara på frågor som användaren har om systemet¹¹⁴.

5.4.2 Fokusgrupper

Denna metod är baserad på information från en grupp på sex till nio användare som tillsammans diskuterar vad de behöver i systemet och hur de upplever det. En MDI-expert har ansvaret för gruppen och leder diskussionen i rätt riktning med hjälp av en förberedd lista över de ämnen som skall diskuteras. MDI-experten skall ha en tillbakadragande roll i gruppen, och trots att denne leder gruppen så skall deltagarna i gruppen känna att diskussionen är flytande och relativt ostrukturerad. Man skall vara noga med att alla deltagare deltar i diskussionen och förhindra att en deltagares synpunkter dominerar diskussionen.

¹¹² Nielsen, Jakob, *Usability engineering* (1993)

¹¹³ Ibid

¹¹⁴ Ibid

Fokusgruppsmetoden frambringar ofta användarnas spontana reaktioner och idéer. Efter diskussionen skriver MDI-experten en kort rapport om hur stämningen var i gruppen och om det var några kritiska kommentarer i diskussionen. Det är en fördel att ha fler än en fokusgrupp då en grupp inte är representativ för alla användare och då diskussionen i en grupp kanske fokuserades på mindre aspekter i systemet. Det är också viktigt att gruppen består av olika typer av användare.

En nackdel med metoden är att den tenderar att ha låg validitet och den är mycket svårt att analysera på grund av att datamaterialet är ostrukturerat då diskussionen varit öppen i gruppen¹¹⁵.

5.4.3 Intervju

Ett enkelt sätt att få reda på om användarna är nöjda med systemet är att fråga dem. I en intervju frågar en MDI-expert helt enkelt användaren vad denne tycker om systemet. MDI-experten kan antingen banta intervjun eller skriva ner det användaren säger. Det första alternativet är kanske att föredra då intervjuaren på så sätt kan koncentrera sig mer på att själva intervjun¹¹⁶.

En intervju kan vara strukturerad i olika hög grad. Detta beskrivs mer utförligt i avsnitt 2.1.3.

5.4.4 Enkät

Vi kommer i detta avsnitt endast att beskriva en typ av enkät som heter "Quis" som är speciellt utarbetad för att mäta användarnas upplevelser av ett system.

Quis, är en engelsk förkortning och står för Questionnaire for User Interaction Satisfaction. Quis är ett mätinstrument som är utarbetat för att mäta hur pass nöjd användaren är med användargränssnittet. Det är framtaget av Human-Computer Interaction Laboratory vid universitetet i Maryland¹¹⁷ och har blivit en enkätstandard när det gäller att fånga upp användarens upplevelser och uppfattning av ett användargränssnitt.

Quis är en av de få kvalitativa metoder som finns till hands för att mäta hur pass nöjda användarna är med systemet. Metoden är relativt billig då den inte är så tidskrävande för användaren. Metoden är bra att använda före och efter det att man har gjort ändringar i användargränssnittet för att på så sätt mäta den eventuella förbättringen.

En nackdel med metoden är att det kan vara svårt för användaren att svara på frågor om hur de upplever användargränssnittet om de inte har det precis framför sig¹¹⁸.

5.5 Prototyping

En prototyp är en experimentell, ofärdig design av ett användargränssnitt. Utvecklandet av prototyper är en viktig del i användarcentrerad design då det

¹¹⁵ Ibid

¹¹⁶ Ibid

¹¹⁷ www.cs.umd.edu/~zzj/UsabilityHome.html

¹¹⁸ Ibid

möjliggör för designteamet att prova sina idéer på användarna och att få feedback från dem¹¹⁹.

Prototyper eliminerar ovissheten om hur väl en design passar användarens behov. Genom prototyping får man synpunkter från användaren vad gäller nödvändig funktionalitet, operationers sekvens, behov av användarstöd, krav på vad och hur saker ska representeras samt användargränssnittets utseende¹²⁰.

Det finns i huvudsak två typer av prototyper; pappersbaserad prototyp och datorbaserad prototyp.

Lo-fidelity prototyp, eller lo-fi, är ett annat ord för pappersprototyp. Med den här metoden konstruerar man en lo-fi modell tidigt i designfasen för att på så sätt ta fram en fungerande prototyp. Genom att göra detta försäkras man sig om att användbarhet byggs in i designen. Modellen implementeras med papper och penna och kan konstrueras snabbt och billigt. Metoden gynnar en iterativ användarcentrerad design. Man testar prototypen på de framtida användarna och samlar in alla åsikter och synpunkter man kan få från dem. Information från användarna utvärderas och prototypen ändras utifrån resultatet från utvärderingen. Denna process, där man testar och omdesignar ett användargränssnitt, repeteras flera gånger under projektets designfas¹²¹.

Vid datorbaserade prototyper kodar man en version av systemet med begränsad funktionalitet så att användaren kan integrera med det¹²². Fördelen med denna metod, till skillnad från lo-fi prototyping, är att användarna får en mer verklig känsla för hur användargränssnittet kommer se ut och hur systemet kommer att fungera. Det finns dock flera nackdelar med en kodad prototyp. En stor nackdel är att den är dyr och tar lång tid att utveckla. Den största delen av den tid som är avsatt för designarbetet kommer att läggas på kodning av prototyp, vilket gör att tiden för att testa prototypen och sen ändra den blir otillräcklig. En annan nackdel är att det finns en risk att användaren inte vågar kritisera ett användargränssnitt som i stort känns klart. Det finns även en risk att användaren luras att tro att systemet snart är färdigt fast det i själva verket är långt kvar till ett färdigt fungerande system¹²³.

Problemen som kan uppstå när man använder sig av en kodad prototyp kan förhindras genom att man istället använder sig av lo-fi modellen. Det finns flera fördelar med denna metod¹²⁴:

- Det går snabbt och enkelt att göra ändringar i användargränssnittet.
- Eftersom lo-fi modellen går så mycket fortare att skapa än en kodad modell, kan den testas mycket tidigare i designfasen. Detta gör i sin tur att mer tid ges till att prova olika idéer och upprepa tester med användare.
- Lo-fi fokuserar på designdelar i systemet istället för implementeringsdetaljer.
- Designers är mindre motvilliga till att göra ändringar i en snabbt gjord pappersmodell än i en prototyp som det tagit flera timmar att koda.

¹¹⁹ Preece, Jenny, *Human-computer-interaction* (1996)

¹²⁰ Ibid

¹²¹ Prototyping for Tiny Fingers, www.cs.mu.oz.au/~tetra/index1.html

¹²² Preece, Jenny, *Human-computer-interaction* (1996)

¹²³ Prototyping for Tiny Fingers, www.cs.mu.oz.au/~tetra/index1.html

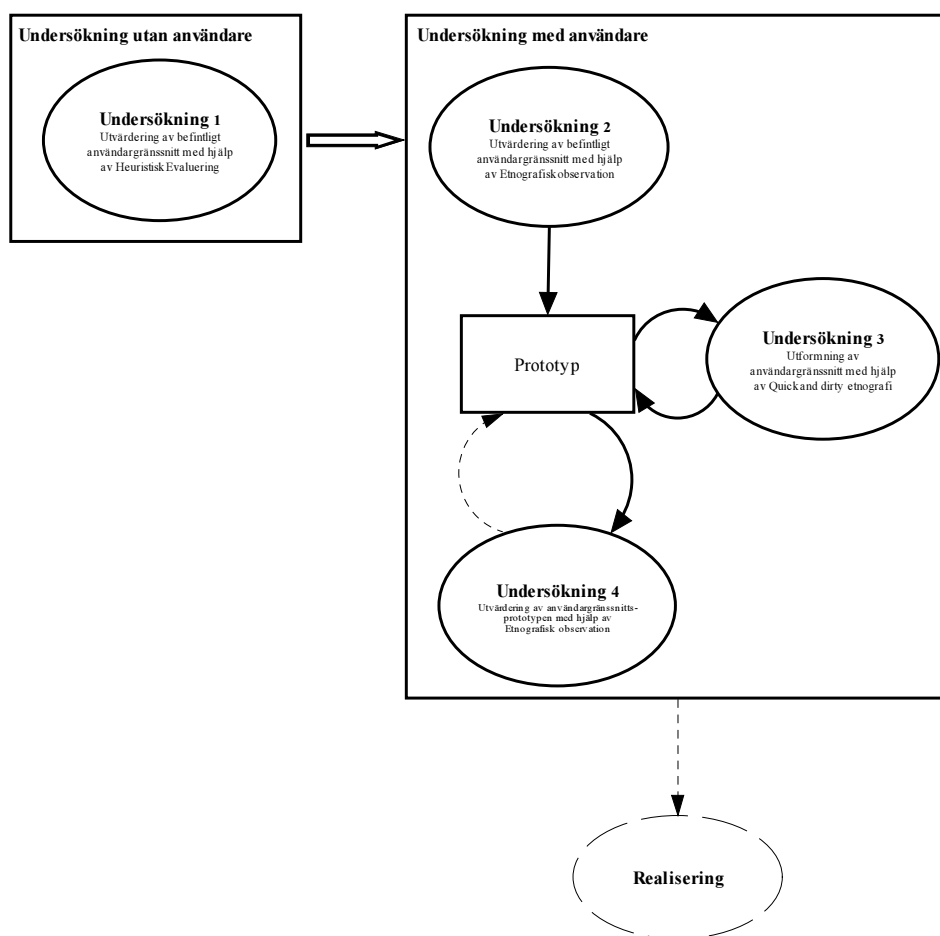
¹²⁴ Ibid

Del III Resultat

6. Ramverk

I det här avsnittet beskriver vi den metod som vi tagit fram utifrån teorin. Metoden bygger dels på de utvärderingsmetoder som vi beskrivit i teoriavsnittet och dels på de forskningsmetoder som vi tagit upp i kapitel 2. Vi har tagit de mest relevanta delarna från dessa metoder och sammafört detta till ”vår” metod vilket utgör det ramverk inom vilket vi har arbetat. Bakgrunden till att vi valde att använda oss av just dessa metoder är att vi hade en diskussion med två användbarhetsexperter på Mind som har praktiskt erfarenhet inom detta området. Tillsammans kom vi fram till att dessa metoder var relevanta för ”vår” metod. Vidare ansåg vi att dessa metoder var praktiskt möjliga att utföra med de resurser vi hade till förfogande.

I kapitel 7 verifierar vi ”vår” metod genom att testa den och beskriver här varje delundersökning och dess resultat för sig. Slutligen sammanfogar vi i kapitel 8 dessa resultat till ett ”Slutligt resultat” där vi besvarar vår problemställning. Vi vill påminna om att vi med ”Systemet” menar det befintliga system som utgör det praktikfall som vår studie bygger på.



Figur 6.1. Vår metod för att öka användbarheten i ett befintligt system

Figuren ovan beskriver vårt tillvägagångssätt i den studie vi gjort. Som figuren visar har vi gjort fyra undersökningar där resultatet från varje undersökning ligger till grund för nästföljande undersökning. Det som är streckat i figuren ovan är sådant som ligger utanför ramen för vår uppsats. Vi har ändå valt att ta med det i figuren för att ge en bild av hur det fortsatta arbetet skulle kunna se ut.

Vi började vår studie med en Heuristisk Evaluering av det befintliga systemet vilket innebar att vi utvärderade användargränssnittet för att finna avvikelser från de tio Heuristiska tumreglerna. Resultatet av denna **första undersökning** var att vi fann ett antal användbarhetsproblem i Systemet. Dessa användbarhetsproblem låg sedan till grund för utformningen av den **andra undersökningen** där vi utvärderade Systemets användargränssnitt tillsammans med användare. I denna undersökning observerade vi och diskuterade med användaren under tiden som denne använde Systemet. Genom att först ha gjort en Heuristisk Evaluering av Systemet hade vi fått en känsla för vilka huvudsakliga problem som fanns i användargränssnittet och visste därmed vilka områden vi skulle fokusera på under observationen och diskussionen med användaren. Resultatet av den andra undersökningen var ett antal användbarhetsproblem sett utifrån användarnas perspektiv. Detta resultat låg sedan till grund för den **tredje undersökning** där vi utformade en prototyp av användargränssnittet. Utformningen av prototypen var baserad på resultatet från de två tidigare undersökningarna. Denna prototyp gjordes på papper, en sk lo-fi prototyp. Vi utvärderade och omdesignade denna prototyp tillsammans med användare tills vi hade fått fram ett användargränssnitt som vi och användargrupperna kände oss nöjda med. Resultatet från denna undersökning var därmed den slutliga användargränssnittsprototypen som vi tagit fram tillsammans med användare.

För att mäta om användbarheten i den slutliga användargränssnittsprototypen ökat i jämförelse med det befintliga användargränssnittet gjorde vi en **fjärde undersökning**. I denna fjärde och sista undersökning utvärderade vi den framtagna användargränssnittsprototypen på ungefär samma sätt som vi utvärderade det befintliga systemets användargränssnitt i undersökning 2. Resultatet från denna fjärde undersökning visade dels om användbarheten i användargränssnittet hade ökat eller ej, och dels på eventuella nya användbarhetsproblem i användargränssnittet som borde åtgärdas för att ytterligare öka användbarheten i systemet.

7. Empiri

Syftet med det här avsnittet är att redovisa den empiri som leder fram till vårt resultat. Vi testar här den metod vi tagit fram och beskrivit i föregående kapitel.

7.1 Undersökning 1

7.1.1 Syfte

Syftet med den första undersökningen var att med hjälp av en Heuristisk Evaluering identifiera avvikelser från evalueringens tio tumregler i det befintliga systemet. Genom att göra detta fick vi en känsla för de användbarhetsproblem som fanns i Systemet. Denna evaluering skulle sedan ligga till grund för utformningen av undersökning 2 där vi utvärderade Systemet tillsammans med användare.

7.1.2 Metod

Den Heuristiska Evalueringen skall egentligen göras av ett flertal oberoende utvärderare. Vi valde dock att göra utvärderingen tillsammans. Detta då ingen av oss hade någon som helst erfarenhet av en sådan här typ av utvärdering. Genom att arbeta tillsammans kunde vi gemensamt diskutera och komma fram till avvikelser i Systemet. En annan orsak till att vi utvärderade Systemet tillsammans var att vi hade begränsat med tid och det skulle ta längre tid att sammanställa två olika utvärderingar än att sammanställa en.

Vi använde oss även av utvärderingsmetoden Standard Inspektion under den Heuristiska Evaluering. Då man på Mind arbetar i Microsofts Windows miljö, jämförde vi Systemet med standarden för Windows.

7.1.3 Arbetsprocedur

Figuren nedan visar Systemets huvudfönster, det sk primära fönstret. Detta fönster kallas i Systemet för "Time Sheet", vi har dock valt att översätta det till "tidsark". Det finns ett tidsark för varje vecka.

Job No.	Job Name	Act. No.	Activity Name	Task	Task Description	Monday	Tuesday	Wednesday	Thursday	Friday
300015	Osberg & Co Ts	101	Installation - time	315	Customer Training	0,0	0,0	0,0	0,0	0,0
300016	Osberg & Co Ts	105	Launch - time	318	Project Meeting	0,0	0,0	0,0	0,0	0,0

Figur 7.1. Tidsark (Time Sheet), det primära fönstret i Systemet

Tidsarket består av en undre och en övre del. I den övre delen finns information om användaren, som exempelvis namn och befattning. Vidare finns där information som t ex vilken vecka det är och den totala arbetstiden för veckan. Den undre delen är själva "inputdelen". Här får användaren fylla i de projekt som hon eller han har jobbat med under veckan samt hur mycket tid som har lagts ner på respektive projekt. Utifrån tidsarket öppnar man andra fönster, sk sekundära fönster, varifrån man hämtar in information till tidsarket. Vi kallar dessa fönster för "sökfönster". Systemet består av totalt fyra sökfönster; sökning efter jobb ("Job"), sökning efter aktivitet ("Activity"), sökning efter uppgift ("Task") och sökning efter ett specifikt tidsark för en särskild vecka.

Vid evalueringen följde vi de tio Heuristiska tumreglerna (se kapitel 5.2.1), samtidigt som vi utgick från teorins generella riktlinjer för design av användargränssnitt (se kapitel 4). Vi började utvärdera tidsarket och fortsatte sedan med sökfönstren.

7.1.4 Evaluering utifrån de tio tumreglerna

1. Använd en enkel och naturlig dialog.

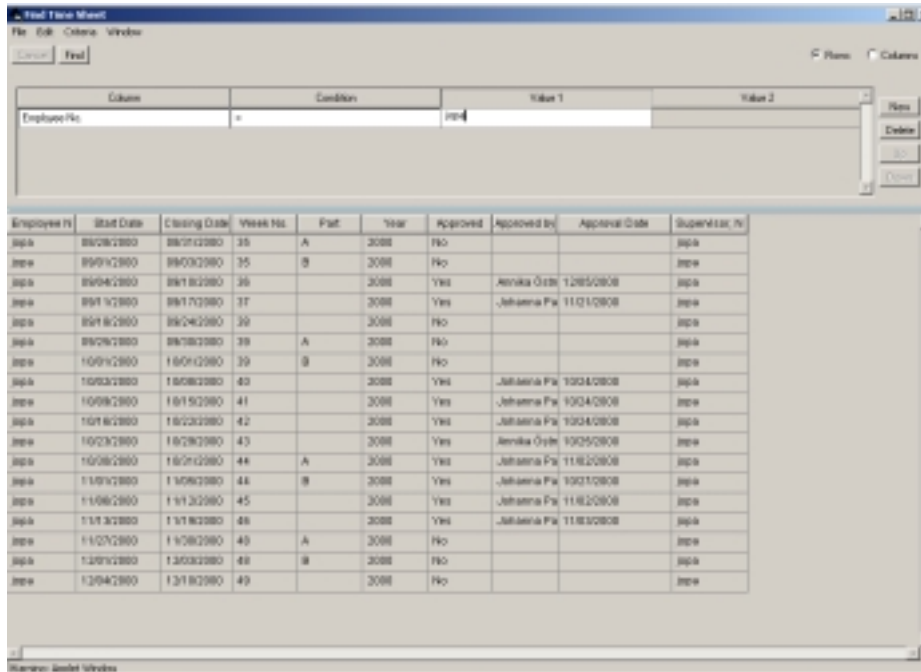
Ett fönster och alla delar i ett fönster skall var organiserat så att det stödjer de uppgifter användaren skall utföra i systemet. I Systemet är informationen allt annat än enkel och naturlig. Exempelvis är det mycket information under menyerna i menyraden som är ologiskt placerad, dels vad gäller under vilken meny informationen ligger och dels informationens placering i menyn. Viss information som man använder ofta ligger exempelvis längst ner i menyerna istället för överst där de är mer lättillgängliga. Det finns även en hel del information under menyerna som vi anser att man aldrig använder sig av. Denna information borde över huvudtaget inte finnas med, då onödig information enligt de generella riktlinjerna bör elimineras.

En annan orsak till att menyraden och dess undermenyer inte känns logiska är att den endast till viss del följer Windowsstandarden. Exempelvis finns menyerna "File" och "Edit" i Systemet, vilka även finns i Windows. Det finns dock ett antal menyer i Systemet som inte existerar i Windows. Vi anser att vissa av dessa menyers undermenyer skulle kunna passa in under "File" och "Edit".

Riktlinjen att onödig information bör elimineras, är något som borde följas även när det gäller informationen som visas i tidsarkets övre del. Här finns en mängd information som man aldrig tittar på och även viss information som man inte ens förstår. Som det är nu måste man skrolla i fönstret för att se viss information.

Enligt den Heuristiska Evalueringens tumregel bör systemets användargränssnitt stödja användarens naturliga sätt att lösa en uppgift. I den undre delen i tidsarket redovisar användaren sin arbetstid, vilket är huvuduppgiften i Systemet. Denna uppgift görs på intet sätt naturligt. Sättet att utföra funktionerna som behövs för att kunna lägga in nya projekt, ta bort projekt, kopiera projekt från tidigare period etc, är många gånger ologiska och omständiga. Även i denna undre del av tidsarket behöver man skrolla för att se all information.

Även i sökfönstren, se figuren nedan, dit man går för att hämta information, finns en hel del onödig information. Det gäller då främst ett antal kolumner som är svåra att förstå innebörden av.



Figur 7.2. Find Time Sheet, ett av sökfönstren i Systemet

I dessa fönster är det meningen att man skall kunna söka efter viss information. Sökfunktionen är dock så komplicerad att det krävs både tid och ansträngning för att förstå den. Det är knappt man förstår att detta är en sökfunktion över huvudtaget. Detta strider mot människors kognitiva förmåga som vi beskrivit tidigare i kapitel fyra; *användaren skall i så liten omfattning som möjligt behöva använda sin intelligens för att lösa en uppgift.*

Färger kan användas för att förenkla tolkning av information och samtidigt göra ett system mer attraktivt och trevligt att använda. I Systemet är alla fönster gråa, vilket gör att det känns tråkigt att arbeta med det.

2. Tala slutanvändarens språk.

Enligt Nielsen skall användargränssnittet helst vara på användarens modersmål. Systemet är på engelska vilket i sig är en avvikelse från tumregeln. Vi anser dock inte detta vara av större betydelse då Mind är ett internationellt företag där det krävs att de anställda behärskar engelska. Vad som däremot är av större vikt är att termer och begrepp bör baseras på användarnas begreppsvärld. I Systemets användargränssnitt finns en mängd begrepp och termer som är svåra att förstå innebörden av. Detta gäller både i menyraden, dess undermenyer samt i själva fönstret. "Action" och "Index" är exempel på sådana begrepp.

I sökfönstren finns en hel del begrepp som är oförståeliga, dels är vissa kolumnnamn i sökresultatet svåra att förstå, dels är namnen på knapparna som utför vissa funktioner svårbegripliga. Exempelvis har knapparna "New" och "Find" inte alls den betydelse man först tror. Sökfunktionen i sökfönstret har svårbegripliga rubriker för de fält man skall fylla i för att söka efter något (se den övre delen av sökfönstret i figur 6.3).

3. Minimera användarens minnesbelastning.

Alla funktioner som Systemet har att erbjuda går att finna i menyerna varför användaren inte behöver komma ihåg någonting utantill. Däremot är menyernas termer svåra och strukturen ologisk vilket bidrar till att det blir svårt att komma ihåg var saker och ting finns.

Många funktioner som utförs i Systemet kräver att de görs i en viss ordning för att det inte skall bli fel. Denna sekvens är inte alltid logisk, varför det i vissa fall kan vara svårt att minnas hur man skall göra.

4. Var konsekvent

Vissa kolumnrubriker i tidsarket är inkonsekventa sinsemellan. Exempelvis en sådan enkel sak som att kolumnen ”Task” borde ha hetat ”Task No” då kolumnerna ”Job No” och ”Act No” är skrivna på det sättet. Vidare är vissa begrepp förkortade på olika sätt mellan de olika fönstren. Ett exempel på detta är att en kolumn i tidsarket heter ”Act No” medan det i sökfönstret för aktivitet heter ”Activity No” för samma kolumn.

5. Ge återkoppling (feedback)

Systemet ger dålig feedback vid flera tillfällen där man skriver in värden. Exempelvis vid tidsredovisningen behöver man fylla i tre fält med värden som sinsemellan har en koppling. Det är dock först när man fyllt i alla fält som man får bekräftat om det man har fyllt i stämmer eller ej.

Ett annat exempel på dålig feedback är vid en sökning i sökfunktionen. Man får då inget meddelande om det man sökt efter inte existerar.

Vidare när man skall godkänna och skicka iväg en veckas tidsredovisning får man ingen bekräftelse på att detta skett utan man får helt enkelt lita på Systemet.

6. Ha tydliga vägar tillbaka

När man öppnar ett sekundärfönster från huvudfönstret är detta fönster lika stort som huvudfönstret, vilket gör det svårt att se om ett helt nytt fönster har öppnats eller om det gamla fönstret har ersatts med ett nytt fönster. Detta gör att man kan känna sig osäker när man skall stänga ner sekundärfönstret då man i vissa fall tror att hela programmet stängs ner. Det finns inga tydliga vägar tillbaka från sekundärfönstret till huvudfönstret.

Det finns ingen ångerfunktion i något fönster som ångrar det senaste man har gjort.

7. Stöd genvägar

Det finns en hel del kortkommandon i Systemet, varav vissa följer Windowsstandard och vissa inte. Vissa kortkommandon kan ibland vara något komplicerade att använda då de endast fungerar beroende på var någonstans i fönstret man står med markören.

Vi saknar ett enklare sätt att hämta in information till huvudfönstret istället för att behöva gå in och söka i sökfönstret då detta tar relativt lång tid.

Systemet saknar ”defaultvärden” vilka hade minskat antalet inmatningar från användaren.

8. *Ge bra felmeddelanden*

Det saknas felmeddelanden på flera ställen i Systemet. Några av de felmeddelanden som finns är förståeliga med de flesta är oklara och förvirrar mer än de hjälper till. Det förekommer även långa textmeddelanden som är så långa och svårbegripliga att man inte orkar läsa dem. Dessa meddelanden dyker upp när man minst anar det och säger ingenting om vad man har gjort för fel.

Få av meddelandena ger konstruktiv hjälp för att lösa problemet.

9. *Förhindra fel*

Mycket av den information (projekt och liknande) som finns att hämta in från ett sökfönster till tidsarket är inaktuell. Ett sätt att förhindra felet man får när man hämtat in ett inaktuellt värde skulle kunna vara att Systemet på något sätt markerar de värden som är aktuella eller helt enkelt tar bort de inaktuella värdena från Systemet.

Ett annat sätt att förhindra fel är att använda sig av fler "defaultvärden" i Systemet.

10. *Ge bra hjälp och dokumentation*

Det finns ingen hjälpdialog i Systemet. Det finns dock en hjälpsida. Men informationen på hjälpsidan är ostrukturerad och det är svårt att hitta det man vill få hjälp med.

7.1.5 Resultat från undersökning 1

Resultatet från den Heuristiska Evalueringen är givetvis alla de punkter med avvikelser från tumreglerna som vi nämnt ovan. Vi har dock valt att dela in dessa i fem huvudgrupper med problem som vi anser vara de mest relevanta för att förbättra Systemets användbarhet:

- | |
|--|
| <ol style="list-style-type: none">1. Ologisk struktur av informationen i menyn2. Svårbegripliga termer3. Dålig feedback vid flera tillfällen4. Svårbegripliga funktioner; bristande förklaringar och svåra begrepp (särskilt vad gäller sökfunktionen)5. För mycket information i fönstret |
|--|

Figur 7.3. Resultat från den Heuristiska Evalueringen

7.2 Undersökning 2

7.2.1 Syfte

Det primära syftet med denna undersökning var att identifiera användbarhetsproblemen i Systemet utifrån användarens perspektiv.

Detta syfte uppnåddes genom att:

- observera och intervjua användaren när denne utförde uppgifter i Systemet.
- samla in användarens attityder till Systemets användargränssnittet genom att låta denne fylla i ett attitydformulär.

- låta användaren ge konkreta förslag på förändringar i Systemet.

7.2.2 Metod

För att vi i undersökningen skulle få så många olika typer av användare som möjligt valde vi ut användare från samtliga fyra enheter på Mind. Då enheterna Technology samt Design & Communication är större än de två andra enheterna valde vi ut två personer från dessa. Vi tog även hänsyn till användarnas olika erfarenhet av Systemet och valde från dessa två enheter ut en nybörjare, dvs en användare som använt Systemet mindre än tre månader, och en användare med längre erfarenhet av Systemet. Från de två resterande enheterna valdes en person per enhet ut. Sammanlagt valdes sex personer ut till detta användartest.

Som nämns i forskningmetodavsnittet tidigare i uppsatsen har vi i denna undersökning använt oss av etnografisk observation. Vi observerade med andra ord användarna under en begränsad tid (ca 45 minuter per användare) när de använde Systemet. Genom denna etnografiska observation kunde vi identifiera de användbarhetsproblem som användaren upplevde med Systemet. Vi fick även användarnas kommentarer om Systemet och förslag på förändringar som borde göras för att öka användbarheten.

Vidare användes en blandning av utvärderingsmetoderna Tänka-högt-protokoll och Coachingmetoden. Tänka-högt-protokoll utfördes genom att vi uppmanade användaren att "tänka högt" medan denne löste uppgifter i Systemet. En av oss satt tillsammans med användaren under hela testet och agerade på så sätt som testledare. Testledaren hade till uppgift att stötta, uppmuntra och samtidigt uppmana användaren att "tänka högt" under tiden denne arbetade med Systemet. Detta kan ses som en typ av Coaching Metod, med den skillnad att vi inte är experter på Systemet och att användaren inte fick ställa frågor om hur uppgifterna skulle lösas under tiden som dessa utfördes. Dessa två metoder passar bra att utföra i samband med observationen då de ger oss information om användarnas synpunkter och reflektioner kring Systemet. Användaren pratar under observationen fritt vilket gör att vi får deras spontana kommentarer, vilka kan vara svåra att få vid enbart observation.

Användarna fick även fylla i ett attitydformulär för att vi på så sätt skulle få en större förståelse för användarnas attityd till Systemet och för att vi på ett enkelt sätt skulle kunna mäta Systemets användbarhet.

Vi avslutade undersökningen med en kortare intervju där användarna fick ge förslag på eventuella förändringar i användargränssnittet.

Innan vi genomförde själva undersökningen gjorde vi ett pilottest med en person på Mind som har god erfarenhet inom MDI-området. Vi fick därigenom tips och råd för hur vi skulle gå tillväga i den riktiga undersökningen.

7.2.3 Testomgivningen och testutrustningen

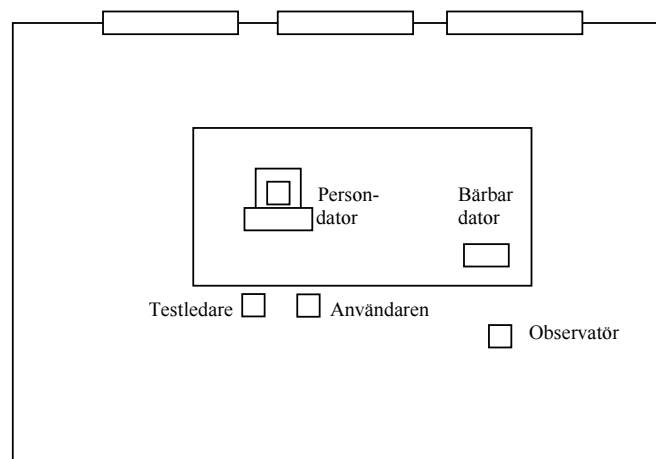
Vi utförde användartestet i ett av Minds konferensrum. Vi valde detta rum då det var rymligt och då det låg avskilt från de som arbetar på Mind. Vi blev på detta sätt inte störda av t ex folk som pratar eller telefoner som ringer. Rubin¹²⁵ menar att det är bra

¹²⁵ Rubin, Jeffrey, Handbook of usability testing (1994)

att utföra användbarhetstester i den miljön som användaren är van att arbeta i. Vi ansåg dock att det skulle vara svårt att utföra testet i den vanliga arbetsmiljön då vi ville att användaren skulle sitta i en lugn miljö eftersom testet var relativt omfattande och tidskrävande. Vi trodde även att användaren skulle få svårt att koncentrera sig om vi utförde testet i användarens arbetsmiljö. En annan orsak var att användaren uppmanades att "tänka högt" under testet vilket skulle kunna kännas obekvämt bland de övriga anställda.

Testrummet såg ut som bilden nedan visar. Vi använde oss av två datorer som båda hade en testversion av Systemets webbapplikation. Vi hade dels en persondator som användaren använde då hon eller han utförde uppgifterna i Systemet, och dels en bärbar dator som var kopplad till den persondator användaren använde. Den bärbara datorn var till för observatören så att hon på ett lättare sätt skulle kunna se vad användaren gjorde i Systemet utan att behöva sitta alltför nära.

Ingen videoupptagningen gjordes då vi ansåg att vi inte skulle få ut något mer av testet genom att använda videokamera. Att videofilma testet tror vi också skulle kunna kännas stressande och onaturligt för användaren. Vi strävade efter att ha en så avslappnad och naturlig miljö som möjligt när vi utförde vårt test.



Figur 7.4. Rummet för undersökning 2

7.2.4 Testledarens och observatörens roll

Testledarens uppgift var att leda hela testet. Det var testledaren som kommunicerade med användaren och som berättade vad denne skulle göra. Testledaren satt bredvid användaren och förde en diskussion under hela testet. Vi upplevde att användaren på så sätt fick en känsla av att testledaren samarbetade vid utförande av uppgifterna vilket kunde kännas lugnande för användaren. Testledaren uppmuntrade användaren att "tänka högt" och kommenterade under tiden användarens handlingar.

Observatörens roll var mer "flugan på väggen", dvs observatören hade inte någon som helst kontakt med användaren under tiden som denne utförde uppgifterna i Systemet. Observatören hade till uppgift att anteckna det som användaren gjorde och sa och satt utom synhåll för användaren. För att underlätta för observatören hade vi utarbetat en mall som innehöll uppgifterna som användaren skulle uträtta och olika tänkbara sätt

att utföra uppgifterna på. På detta sätt kunde observatören enkelt kryssa i det tillvägagångssätt som användaren valde och kunde därmed fokusera mer på användarens kommentarer och eventuella problem som denne hade med Systemet. Detta kan ses som en typ av fältobservation vilken vi beskrivit i avsnitt 5.4.1 där observatören inte på något sätt stör användaren och där observatören använder sig av en förberedd lista på det som skall observeras.

7.2.5 Testprocedur

7.2.5.1 Introduktion till testet

Innan användaren kom till testlokalen hade vi loggat in på webbapplikationens testmiljö. Användaren fick härifrån utföra ett antal uppgifter i Systemet.

När användaren anlände till testlokalen hände följande:

- Användaren hälsades välkommen
- Testets innehåll beskrevs övergripande

Följande information gavs:

- Användarens anonymitet är skyddad
- Användaren kan när som helst avbryta testet
- Eventuella problem med hanteringen av Systemet som användaren upplever beror på en dålig utformning av användargränssnittet och inte på användaren själv

Följande uppmaningar gavs:

- Användaren måste vara ärlig och uppriktig i sina kommentarer
- Användaren uppmanas att läsa varje uppgift högt och förklara kort hur denne uppfattar frågan
- Användaren uppmanas att "tänka högt" när denne löste uppgifterna i Systemet

7.2.5.2 Testets innehåll

Testet bestod av tre delar:

- Uppgifter som användaren skulle utföra i Systemet
- Attitydformulär
- Avslutande intervju

7.2.5.3 Uppgiftslista

Uppgifterna var utformade så att användarna bara kunde se en uppgift i taget. Uppgifterna innehöll vanliga aktiviteter som användarna brukade göra i Systemet så som att redovisa arbetstidentiden, söka efter ett speciellt jobb etc. Se bilaga B1.

7.2.5.4 Attitydformulär

Attitydformulärets främsta uppgift var att mäta användarens attityd till Systemet. Detta är främst ett komplement till observationen och intervjun och är egentligen inte en enkät i den bemärkelsen att den delas ut till 30 personer, utan endast till de som medverkar i testet. I en skala från 1-5 fick användarna svara på påståenden. Påståendena belyste användarnas synpunkter vad gäller exempelvis Systemets information och dess placering, felmeddelanden samt Systemets funktioner. Se bilaga B2. Attitydformuläret bygger på enkätstandarden Quis som vi beskrivit i avsnitt 5.4.4.

7.2.5.5 Avslutande intervju (diskussion)

Efter det att användaren gjort alla uppgifter i Systemet och svarat på attitydformuläret hölls en avslutande diskussion med användaren. Denna diskussion inleddes med en fråga till användaren (se bilaga B3). Syftet med diskussionen var att diskutera de eventuella problem med användargränssnittet som användaren upplevde i Systemet. Användaren fick även möjlighet att ge förslag till förbättringar av användargränssnittet.

7.2.6 Resultat från undersökning 2

Vi redovisar resultatet genom att ta upp de användbarhetsproblem som enligt vår tolkning är de allvarligaste utifrån användarnas perspektiv. Vi börjar med att redovisa resultatet från observationen, dvs när användaren utförde uppgifter i Systemet. Trots att vi i testet gjorde attitydformuläret före den avslutande intervjun har vi valt att redovisa resultatet från attitydformuläret sist. Detta då vi anser att observationen och den avslutande intervjun är starkt kopplade, vilket gör det lämpligt att beskriva dem i samband med varandra.

Under observationerna såg vi att det fanns stor skillnad på hur väl användarna klarade uppgifterna i Systemet. Vissa användare visste nästan direkt hur de skulle lösa en uppgift medan andra var osäkra och fick leta efter funktioner och kommandon i Systemet för att lösa uppgiften. Två användare körde under testet fast totalt och fick hjälp av oss för att kunna fortsätta med uppgifterna.

Gemensamt för användarna var att alla mer eller mindre fick leta i menyerna för att hitta det de sökte efter. Som en användare uttryckte det; "Detta system följer ingen standard över huvudtaget vad gäller termer och dess placering. Allt är helt ologiskt". En annan användare tog själv på sig skulden då denne inte hittade i Systemet; "Det är nog mitt dåliga minne som gör att jag inte kommer ihåg var saker och ting ligger".

Det märktes också att användarna är vana vid Windowsstandard. När de till exempel skulle ta bort en rad i en tabell så försökte de klicka i början av raden för att på så sätt markera den och sedan trycka på "delete"-knappen på tangentbordet. Så fungerar ju exempelvis Excel i Windows, dock fungerar det inte så i Systemet.

En positiv iakttagelse vi gjorde var att användarna inte var rädda för att prova funktioner eller kommandon som de aldrig tidigare hade provat. En anledning till detta kan vara att de visste att detta endast var en testversion av Systemet och att det därför inte gjorde något om de gjorde fel.

Nedan följer en sammanställning från observationen.

- | |
|---|
| <ol style="list-style-type: none">1. Får leta i menyerna för att hitta funktionerna2. Osäker om funktionen gjorde det som den var tänkt att göra. Dålig feedback3. Förstår ej hur de skall använda funktionerna4. Hittar ej funktionerna5. Ologiska namn både i menyerna och i övriga delar av Systemet |
|---|

Figur 7.5. Resultat från observationen i undersökning 2

Som figuren ovan visar kom vi i observationen fram till ett antal användbarhetsproblem i Systemet. Vi har rangordnat problemen med det allvarligaste

problemet först. Denna rangordning av problemen är utifrån vår tolkning av observationen. Ju fler av användarna som hade samma problem, desto allvaligare anser vi problemet vara.

Det största användbarhetsproblemet var att användarna fick leta i menyerna för att hitta funktionerna. Vidare ser vi att användarna var osäkra på om funktionen gjorde det som de trodde att den skulle göra, dvs feedbacken i systemet var dålig. Andra problem med Systemet var att användarna inte visste hur de skulle använda funktionerna samt att det var svårt att hitta dem.

Om vi går vidare till den avslutande intervjun där vi ställde frågan; "Vad anser du vara det största problemet med Systemets användargränssnitt, finns det något du skulle vilja ändra på?" skiljer sig svaren något från resultatet i observationen. Det naturliga hade kanske varit att användarna skulle vilja ändra på sådant i Systemet som de hade haft problem med när de utförde uppgifterna. Detta var dock inte alltid fallet. Under observationen var det största användbarhetsproblemet enligt vår tolkning att de fick leta i menyerna för att hitta funktionerna. Resultatet från intervjun visade att det största användbarhetsproblemet var att det var ologiska namn både i menyerna och i övriga delar av Systemet. Dessa två användbarhetsproblem har i och för sig en viss koppling till varandra. Hur som helst fick vi, genom att både observera och intervjua användaren, en bredare förståelse för Systemets användbarhetsproblem.

Även fast vi i den avslutande intervjun ställde en semistrukturerad fråga till användaren liknade denna del av undersökningen mer en diskussion än en intervju. Beroende på vilket svar vi fick på frågan kunde vi därefter ställa följdfrågor och på så sätt skapa en diskussion med användaren. Exempel på en användares kommentar kring Systemet var; "Färgerna är otroligt tråkiga. Systemet är tråkigt som det är och den gråa färgen gör det inte roligare att använda ". En annan användare sa; "Allt är bara grått". Samma användare menade att "det känns som om de har glömt att ta med en kognitionsvetare i systemutvecklingen. Systemet är nog bara byggt av tekniker".

Resultatet från den avslutande intervjun ser ut på följande vis:

- | |
|---|
| <ol style="list-style-type: none">1. Ologiska namn både i menyerna och i övriga delar av Systemet2. Mycket onödig information3. Informationen i menyerna ligger ologiskt4. För få "defaultvärden" i Systemet |
|---|

Figur 7.6. Resultat från den avslutande intervjun i undersökning 2

Rangordningen av användbarhetsproblemen i figuren ovan är, liksom resultatet från observationen, gjord utifrån vår tolkning. Figuren ovan visar att användarna tycker att Systemet använder sig av ologiska termer. Vissa användare ansåg även att de engelska termerna i Systemet kunde vara svåra att förstå.

Användarna ansåg även att det fanns en hel del onödig information i användargränssnittet. Det var mycket information i Systemet som användaren aldrig tittade på. Det tredje användbarhetsproblemet var att informationen i menyerna låg ologiskt vilket gjorde att de fick leta onödigt mycket för att finna det de sökte efter. Det sista användbarhetsproblemet var att användargränssnittet saknade

”defaultvärden”. Ofta får användaren skriva in samma information varje gång de skall tidsredovisa, vilket enligt användarna skulle kunna lösas genom ”defaultvärden”.

Går vi vidare till attitydformuläret, som är det sista deltestet i denna undersökning, såg resultatet ut på följande vis:

- | |
|--|
| <ol style="list-style-type: none">1. Användargränssnittet har fula färger (1,25)2. Användargränssnittet är svårt att överskåda (1,5)3. Felmeddelanden klargör inte problemet och ger förslag på lösning (1,5)4. Vägen till önskad information är inte självklar (1,66)5. Att korrigera ett misstag är svårt (1,83)6. Att använda sökfunktionen är svårt (2,0) |
|--|

Figur 7.7. Resultat från attitydformuläret i undersökning 2

Siffran i parenteserna i figuren ovan är medelvärden av svaren på den specifika frågan. Attitydformuläret hade en skala från 1-5, ju lägre siffra desto mer negativ var användaren. Man bör vara medveten om att dessa siffror är mycket ungefärliga och att de bara ger en känsla för hur allvarligt användarna uppfattade problemen i Systemet.

Som figuren ovan visar är det största användbarhetsproblemet enligt attitydformuläret färgerna i användargränssnittet. Vidare ansåg användarna att användargränssnittet var svårt att överskåda och att felmeddelandena varken klargjorde problemet eller gav förslag på lösning.

Resultatet från denna undersökning, dvs resultatet från observationen, den avslutande intervjun och attitydformuläret, ligger till grund för undersökning 3 där vi skapar en prototyp av användargränssnittet.

7.3 Undersökning 3

7.3.1 Syfte

Syftet med denna undersökning var att skapa en prototyp av Systemets användargränssnitt som kännetecknas av hög användbarhet.

Detta syfte uppnåddes genom att:

- utgå från resultaten från de tidigare undersökningarna
- vi i en iterativ process med användarna diskuterade oss fram till ett förslag på användargränssnittets utformning.
- vi samtidigt tog hänsyn till de riktlinjer som finns vad gäller användargränssnittsdesign.

7.3.2 Metod

Vi använde oss i denna undersökning av utvärderingsmetoden Samupptäckar-metoden (Co-discovery-learning), vilken beskrivs i kapitel 5.3.2. Denna metod innebär att två användare tillsammans försöker lösa ett problem. Anledningen till att vi valde denna metod är att vi tror att det är mer naturligt för användarna att kommentera och komma fram till förbättringar i gränssnittet om man för en diskussion sinsemellan än om man är själv.

Vi valde att göra användargränssnittsprototypen på papper, sk lo-fi prototyping, p.g.a. de fördelar som nämnts i kapitel 5.5.

Den användargränssnittsprototyp som vi presenterade för den första användargruppen var utformad dels utifrån den information vi hade fått från de tidigare undersökningarna och dels utifrån de generella riktlinjerna för hur man designar ett användargränssnitt. Den första användargruppen fick studera vårt prototypförslag och fick sedan göra några uppgifter på prototypen som till viss del liknade uppgifterna i undersökning 2. Användarna fick även ge förslag på ändringar vilka gjordes direkt på pappersprototypen. Innan vi träffade nästa användargrupp omdesignades användargränssnittsprototypen utifrån den förra gruppens önskemål och förslag. Genom att gå fram och tillbaka mellan olika användargrupper kom vi till slut fram till en användargränssnittsprototyp som de flesta var nöjda med. Det bör påpekas att vi inte enbart gick efter användarnas synpunkter vid omdesignen mellan grupperna. Vi tog som tidigare nämnt även hänsyn till riktlinjerna för användargränssnittsdesign.

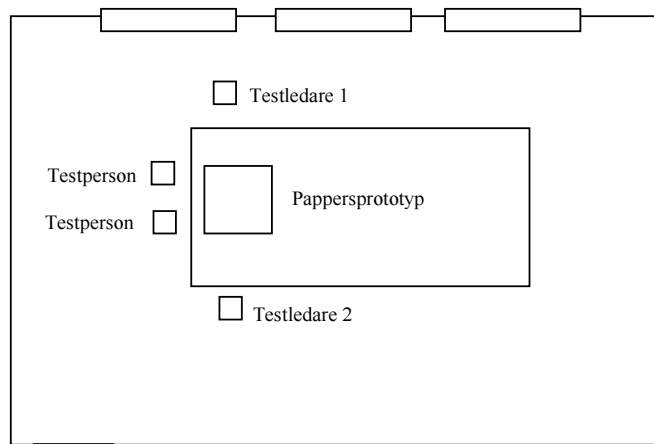
Denna typ av undersökning kallar Hughes för "quick and dirty ethnography"¹²⁶. Undersökningen innebär i vårt fall att man gör fältstudier där man bekantar sig med användaren och dess omgivning. Man skapar sedan en prototyp av systemets användargränssnitt, går ut och gör mer fältstudier, omdesignar användargränssnittet, gör mer fältstudier osv tills dess att man kommit fram till en tillfredsställande prototyp av användargränssnittet.

Innan vi träffade den första gruppen utfördes ett pilottest på en användare. Då detta möte resulterade i förslag på ändringar i användargränssnittet tog vi med dessa ändringar inför mötet med den första användargruppen.

7.3.3 Testomgivningen och testutrustningen

Även detta test utfördes i ett av Minds konferensrum. Denna undersökning krävde emellertid att vi hade lite mer utrymme då två användare var delaktiga i varje test. Som figur 7.8 visar satt användarna bredvid varandra och vi på varsin sida om dem. Prototypen låg framför användarna för att de lätt skulle kunna göra eventuella ändringar. Den enda utrustning vi hade till detta test var ett förslag på en prototyp av användargränssnittet som vi gjort på A3-ark. Vidare hade vi diverse färgpennor och kladdpapper som användarna kunde använda för att göra ändringar på prototypen.

¹²⁶ Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design* (1994)



Figur 7.8. Rummet för undersökning 3

7.3.4 Testledarnas roller

Till skillnad från undersökning 2, där vi hade en testledare och en observatör, hade vi i denna undersökning två testledare. De två testledarna hade dock lite olika arbetsuppgifter under testet. Den ena testledaren, som vi kallar testledare 1, hade huvudansvaret för användarna. Det var denna testledare som i första hand kommunicerade med användarna och hade till uppgift att leda användarna genom testet. Denna testledare informerade även användarna, när de anlände till testlokalen, om testets upplägg och vad testet gick ut på.

Även den andra testledaren, testledare 2, kommunicerade med användarna men i mindre utsträckning än testledare 1. Testledare 2 hade till huvuduppgift att ”agera dator”, dvs ha hand om alla fönster och menyer i användargränssnittet. Klickade t ex användarna på en knapp för att söka, hade testledare 2 till uppgift att lägga fram sökfönsterprototypen åt användarna. Testledare 2 hade även till uppgift att anteckna användarnas förslag på ändringar i användargränssnittsprototypen om inte användarna själva gjorde detta direkt på pappersprototypen.

7.3.5 Testprocedur

7.3.5.1 Introduktion till testet

Innan användarna kom till testlokalen hade vi lagt i ordning pappersprototyperna så att fönstret för inloggning låg överst på användarnas plats. Användarna fick därifrån utföra ett antal uppgifter på pappersprototypen och komma med förslag på förändringar.

När användarna anlände till testlokalen hände följande:

- Användarna hälsades välkomna
- Testets innehåll beskrevs övergripande

Följande information gavs:

- Användarnas anonymitet är skyddad
- Förklarade hur de skulle arbeta ihop under testets gång
- Viktigt att användarna säger vad de tycker om användargränssnittsprototypen och ger seriösa förslag på eventuella förändringar

Följande uppmaningar gavs:

- Användarna måste vara ärliga och uppriktiga i sina kommentarer
- Användarna uppmanas att diskutera högt när de löser uppgifterna på pappersprototyperna
- Användarna uppmanas att noggrant studera användargränssnittets utformning

7.3.5.2 Testets innehåll

Testet bestod av två delar:

- Uppgifter som användarna tillsammans skulle lösa på användargränssnittsprototypen
- Diskussion kring användargränssnittet och dess utformning

7.3.5.3 Uppgiftslista

Användarna fick utföra ett antal uppgifter på pappersprototyperna. Uppgifterna innehöll vanliga aktiviteter som användarna brukade göra i Systemet och som samtidigt testade användargränssnittets utformning. Uppgifterna gavs muntligen av testledare 1. Se bilaga C1.

7.3.5.4 Diskussion

Diskussionen gick ut på att få användarnas kommentarer och synpunkter på användargränssnittsprototypen. Vi diskuterade användargränssnittets olika delar tillsammans med användarna. Bilaga C2 visar de stödord, och därmed de diskussionspunkter, som vi testledare utgick ifrån under diskussionen.

7.3.6 Resultat från undersökning 3

Resultatet av denna undersökning är den slutliga användargränssnittsprototyp som vi fick fram efter det sista mötet med användare. I detta avsnittet beskrivs hur vi kom fram till detta resultat genom att ge exempel på de förslag på ändringar i användargränssnittet som vi fick från varje möte med användare.

I figuren nedan visas den första prototypen vi tog fram utifrån resultaten från de två föregående undersökningarna. Som figuren visar har det gjorts ändringar på prototypen för hand, vilket gjordes tillsammans med testpiloten vi träffade innan den första användargruppen. Figuren är en prototyp av huvudfönstret, vilket kallas för "Time Sheet" i Systemet. Vi har dock valt att kalla det för "tidsark".

Figur 7.9. Första prototypen av tidsarket, eller "Time Sheet"

De största skillnaderna mellan detta användargränssnitt och det befintliga användargränssnittet (se bilaga A1) är att vi i prototypen har ändrat strukturen och en del begrepp i menyn, eliminerat onödig information, gjort om sökfunktion, lagt till en veckoalmanacka i huvudfönstret, infört "defaultvärden" samt sett till att användaren får feedback på de flesta aktiviteterna i systemet.

Menyn förändrade vi bl.a. på så sätt att vi tog bort svårbegripliga menyer så som "Index" och "Action" och tog även bort en hel del onödiga funktioner som låg under dessa och andra menyer. Vi flyttade flera funktioner till "Edit" och lade även till en meny; "Submit". Vi lade till funktionen "undo" under "Edit" så att användaren alltid skall kunna ångra det senaste som gjorts. Vi ändrade även namn på flera funktioner så att dessa skulle bli mer begripliga. Exempelvis ändrade vi namnet på funktionen "Enter Time Sheet Line" till "Insert Time Sheet Line".

Resultatet från undersökning 2 visade att i stort sett inga användare använde sig av information längst ner i huvudfönstrets övre del. För att användaren skall kunna komma åt denna information måste hon eller han skrolla i fönstret. Vi valde därför att ta bort denna information helt och på så sätt slippa skrollen. Den horisontella skrollen på den nedre delen av huvudfönstret togs bort genom att vi förkortade veckonamnen och på så sätt kunde göra kolumnerna smalare.

Veckoalmanackan som vi lade till i huvudfönstret har till syfte att ge en överblick över de föregående veckorna för att se att dessa är godkända. Den erbjuder även en genväg om man vill ta sig till ett tidsark för en viss vecka. Genom att dubbelklicka på ett veckonummer öppnar sig tidsarket för den valda veckan.

Vi erbjuder även andra genvägar i fönstret: T ex funktionen "Submit Time Sheet", som innebär att användaren godkänner och skickar iväg veckans tidsark, har lagts till i

fönstret i form av en knapp. På detta sätt slipper användarna gå upp i menyn för att finna funktionen. Funktionen finns dock även uppe i menyn, under "Submit". Vi har även lagt in genvägar för funktionerna "Find Job" och "Find Time Sheet" vilket innebär att användarna direkt kommer till dessa sökfönster. Genvägen till "Find Time Sheet" insåg vi dock, tillsammans med testpiloten, var onödig då veckoalmanackan är ett enklare sätt att nå samma resultat.

Resultatet i undersökning 1 och 2 visade att sökfönstren var krångliga att använda och att sökfunktionen i dessa fönster var svårbegriplig. Vi såg därför detta som en viktig del att ändra i användargränssnittet. Det finns tre typer av sökfönster där användaren kan finna information om olika jobbprojekt; "Find Job", "Find Activity" och "Find Task". I dagens användargränssnitt måste denna information hämtas en i taget från de olika sökfönstren till huvudfönstret, eller så kan användaren själv mata in dessa värden direkt i huvudfönstret. Vi valde i användargränssnittsprototypen att göra detta till en sekvens där användaren hämtar informationen från de tre sökfönstren i en ordningsföljd utan att komma tillbaka till huvudfönstret mellan varje sökning.

I figuren nedan visas användargränssnittsprototypen "Find Job" som är ett sökfönster där användaren kan söka på ett speciellt jobb eller projekt som denne vill redovisa sin tid på. Användaren kan lätt söka efter ett speciellt jobb i sökfönstret genom att fylla i antingen "Job No" eller "Job Name" i sökrutan. När användaren hittat det sökta värdet dubbelklickar denne på det valda jobbet och kommer på så sätt vidare till ett liknande fönster som heter "Find Activity". Det jobb som man valt i det föregående fönstret kommer då upp i rutan under "Your Choices" i det nya sökfönstret. Användaren gör på samma sätt i detta fönster för att söka efter ett visst värde. När användaren valt det sökta värdet i detta fönster kommer hon eller han slutligen till det sista sökfönstret, "Find Task". (Denna sekvens visas i den färdiga prototypen i bilagorna C8– C10.) När användaren valt informationen i det sista fönstret kommer denne tillbaka till huvudfönstret där värdena då har lagts in i huvudfönstrets tidsredovisningsdel. Är det någon information i denna sekvens som användaren redan vet kan värdet skrivas in manuellt i respektive ruta under "Your Choices" i sökfönstren. Självklart kan användaren även skriva in värdena för "Job", "Activity" och "Task" direkt i huvudfönstret om hon eller han på förhand vet dessa värden.

Figur 7.10. Sökfönstret "Find Job"

Andra ändringar som har gjorts men som inte syns på pappersprototypen är exempelvis att användaren efter inloggning kommer direkt till den nuvarande veckan, och inte som i det befintliga systemet där användaren kommer till tidsarket för den första veckan. En annan ändring är att användaren alltid kommer till sin egna profil efter att denne har loggat in, vilket inte alltid är fallet i dagens system. Eftersom användaren ofta arbetar med samma projekt under flera veckors tid har vi valt att lägga in föregående veckas projekt som "defaultvärden" i det nya tidsarket. En annan ändring är att man kan markera en hel rad i huvudfönstrets tidsredovisningsdel genom att klicka längst ut till vänster på raden. Användaren kan på så sätt enkelt ta bort en rad genom att trycka på "delete"-tangenter på tangentbordet. Vid en sådan, och andra liknande händelser, får användaren alltid feedback i form av en fråga i ett "pop-up"-fönster. Vi har sett till att användaren får feedback på de allra flesta händelser i systemet. Detta gör att användaren känner sig säkrare på att de uppgifter denne gjort i systemet verkligen har utförts.

Resultat efter första användargruppen:

Efter att ha träffat den första användargruppen ändrade vi användargränssnittet utifrån användarnas förslag (se bilaga C4):

- Vi tog bort genvägen till "Find Job" då denna funktion, enligt användarna, inte behövde vara en genväg. Istället gjordes veckokalendern större.
- Menyn "Submit" i menyraden togs bort då det endast fanns en funktion under denna meny. Denna funktion lades istället under "Edit".
- Vissa kolumner i sökfönstren togs bort då användarna ansåg att dessa var onödiga (se figur 7.10).

Andra förslag vi fick från användargruppen var exempelvis färgsättning på användargränssnittet, vilket skulle göra användargränssnittet roligare att arbeta med. De två användarna kom även med förslaget att använda sig av högerklick på musen

för att därifrån kunna använda funktioner som finns under menyn "Edit" såsom ångra, kopiera, radera etc.

Resultat efter andra användargruppen:

Den andra gruppen vi träffade kom med förslaget att man skulle ha en "pull-down"-meny i varje fält i huvudfönstrets undre del, dvs i tidsredovisningsdelen (se bilaga C5). I dessa "pull-down"-menyer skulle användaren direkt kunna välja värdena för "Job", "Activity" och "Task". Genom "pull-down"-menyer slipper användaren gå till sökfönstren för att hämta värdena, utan kan hämta dessa direkt i huvudfönstret. Vi diskuterade tillsammans med användarna fram den bästa lösningen på detta.

Användargruppen kom även med förslaget att man skulle ha möjlighet att skriva in flera rader i "Remarks"-fältet i huvudfönstrets tidsredovisningsdel.

Efter att ha träffat denna grupp ändrade vi även sättet som sökfönstren visades. Istället för att sökfönstren helt skulle ersätta huvudfönstret tyckte användarna att sökfönstren enbart skulle ersätta den övre delen av huvudfönstret. Sökfönstren skulle i övrigt se ut och fungera likadant som tidigare.

Resultat efter tredje användargruppen:

Detta var den sista användargruppen vi träffade. Denna grupp gav som förslag att sökfönstret skulle visas som ett "pop-up"-fönster ovanpå huvudfönstret för att på så sätt verkligen markera att ett nytt fönster öppnats. Det kan annars vara svårt att uppfatta att endast en del av fönstret "byts ut".

Eftersom vi i undersökning 4 ville utvärdera den slutliga prototypen valde vi att göra om den handgjorda pappersprototypen till en kodad prototyp. Detta gjorde vi för att det skulle bli mer tydligt för användaren och för att användaren skulle få en mer "verklig" känsla över hur användargränssnittet skulle komma att se ut i systemet. Den slutliga kodade prototypen av tidsarket visas nedan (se bilagorna C6-C10 för det slutliga resultatet från undersökning 3).

Job No.	Job Name	Act No.	Activity Name	Task No.	Task Description	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Remarks
1	30002	102	Definition	303	Test	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	30003	104	Customize	305	Project Meeting	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	30005	101	Project Meeting	301	Production	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4													

Figur 7.11. Resultat från undersökning 3, den slutliga användargränssnittsprototypen av tidsarket

7.4 Undersökning 4

7.4.1 Syfte

Det primära syftet med detta test var att utvärdera den slutliga prototypen av användargränssnittet som vi tillsammans med användarna utformade i undersökning 3.

Detta syfte uppnåddes genom att:

- Observera och intervjua användaren när denne utförde uppgifter på prototypen.
- Samla in användarens attityder till det nya användargränssnittet genom att låta denne fylla i ett attitydformulär.

7.4.2 Metod

Som nämnts tidigare fick vi av praktiska skäl göra denna sista undersökning på ett annat företag än Mind. Detta företag använde sig också av Systemet för tidsredovisning med den skillnaden att de använde en svensk version.

Sammanlagt var det fyra personer som deltog i denna sista undersökning.

I detta test använde vi oss, liksom i undersökning 2, av etnografisk observation. På samma sätt som vi gjorde i undersökning 2 observerade vi användaren när denne löste ett antal uppgifter i användargränssnittet. Till skillnad från undersökning 2, där vi observerade användarna när de utförde uppgifter i det befintliga systemet, observerade vi här användarna när de utförde uppgifter på

användargränssnittetsprototypen. Då den kodade prototypen saknade funktionalitet valde vi att skriva ut prototypen på papper och därefter göra utvärderingen.

Genom att använda den etnografiska observationen kunde vi på ett relativt snabbt sätt utvärdera den prototyp som vi gjort av användargränssnittet. Vi fick även användarnas kommentarer om användargränssnittet och förslag på förändringar för att ytterligare öka användbarheten.

I likhet med undersökning 2 använde vi i denna undersökning en blandning av utvärderingsmetoderna Tänka-högt-protokoll och Coaching Metoden. Tänka-högt-protokollet användes i denna undersökning genom att vi uppmanade användaren att "tänka högt" medan denne löste uppgifterna på prototyperna. En av oss satt bredvid användaren för att på så sätt stötta, uppmuntra och uppmäna användaren "tänka högt". Även då vi i detta fall inte arbetade med ett riktigt datorsystem, anser vi att detta är en typ av Coaching Metod. Vi kan här till skillnad från undersökning 2 utnämna oss till experter av systemet, även då systemet endast finns i pappersform.

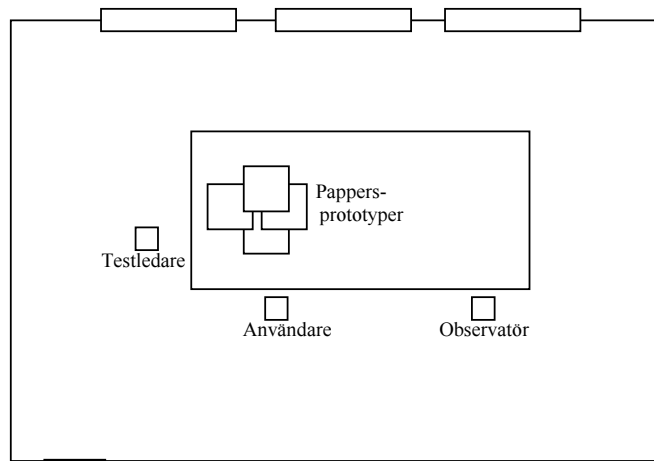
I jämförelse med undersökning 2 blev det i denna undersökning mer naturligt att diskutera kring användargränssnittet och jämföra det med det befintliga systemet. Då utvärderingen utfördes på pappersprototyp gick det relativt långsamt för användarna att utföra uppgifterna. Detta medförde att det blev naturligare att diskutera de olika alternativ till lösningar som fanns till uppgifterna.

Användarna fick även fylla i ett attitydformulär för att vi skulle få en större förståelse för användarnas attityd till användargränssnittsprototypen. Detta attitydformulär liknade i stort det attitydformulär som användes i undersökning 2 för att vi på så sätt skulle kunna mäta användbarheten och jämföra resultaten.

7.4.3 Testomgivningen och testutrustningen

Vi utförde användbarhetstestet i ett av företagets konferensrum. Testrummet såg ut som bilden nedan visar. Det enda material som vi använde oss av under testet var A4 papper med kodade användargränssnitt. För att göra det så verklighetstroget som möjligt gjorde vi huvudfönstret i prototypen i A4-format och "pop-up"-fönstren, dvs sökfönstren, i ett något mindre format. "Pull-down"-menyer var utklippta så att de skulle passa in i prototypen. Även meddelanderutor var formade så att de skulle passa in i "papperssystemet".

Vi valde att göra pappersprototyperna i svartvitt, då val av färger till ett användargränssnitt är en vetenskap i sig och en mycket komplex uppgift. Vi har därför uteslutit färger i användargränssnittet för att inte göra misstag med färgsättningen.



Figur 7.12. Rummet för undersökning 4

7.4.4 Testledarens och observatörens roll

Testledaren hade till uppgift att leda hela testet. Det var testledaren som kommunicerade med användaren och som berättade vad denne skulle göra. Testledaren satt bredvid användaren och förde en diskussion under hela testet. Liksom i undersökning 2 upplevde vi att användaren på så sätt fick en känsla av att testledaren samarbetade vid utförande av uppgifterna. Testledaren uppmuntrade användaren att "tänka högt" och kommenterade under tiden användarens handlingar. Till skillnad från undersökning 2, hade testledaren även rollen som expert på systemet. Testledaren kunde bl.a. ge förslag på alternativa lösningar till uppgifterna. Då användargränssnittet var på papper var det ibland svårt för användaren att se alla händelser och delar i användargränssnittet. Detta medförde att testledaren vid behov förklarade delar av gränssnittsprototypen.

Observatörens roll var inte lika strikt som i undersökning 2. I undersökning 2 hade observatören mer rollen som "flugan på väggen" medan observatören i denna undersökning vid behov kunde flika in med kommentarer och frågor till användaren. Det var dock i första hand testledaren som hade huvudansvaret för diskussionen. Observatörens främsta uppgift var att anteckna det som användaren gjorde och sa under testet.

7.4.5 Testprocedur

7.4.5.1 Introduktion till testet

Innan användaren kom till testlokalen hade vi lagt i ordning prototyperna så att fönstret för inloggning låg överst på användarens plats. Användaren fick därifrån utföra ett antal uppgifter på prototyperna.

När användaren anlände till testlokalen hände följande:

- Användaren hälsades välkommen
- Testets innehåll beskrevs övergripande

Följande uppmaningar gavs:

- Användaren måste vara ärlig och uppriktig i sina kommentarer

- Användaren uppmanas att läsa varje uppgift högt och förklara kort hur denne uppfattat frågan.
- Användaren uppmanas att "tänka högt" när denne löser uppgifterna på pappersprototypen.

7.4.5.2 Testets innehåll

Testet bestod av två delar:

- Uppgifter som användaren skulle utföra på pappersprototyperna
- Attitydformulär

7.4.5.3 Uppgiftslista

Uppgifterna var utformade så att användarna bara kunde se en uppgift i taget. Uppgifterna innehöll, liksom i undersökning 2, vanliga aktiviteter som användarna brukade göra i Systemet så som att redovisa arbetstidentiden, söka efter ett speciellt jobb etc. Se bilaga D1.

7.4.5.4 Attitydformulär

Attitydformulärets främsta uppgift var att mäta användarens attityd till det nya användargränssnittet för att på så sätt kunna göra en jämförelse med resultatet från undersökning 2. Se bilaga D2.

7.4.6 Resultat från undersökning 4

Uppgifterna som användaren fick utföra på pappersprototypen var ungefär de samma som i undersökning 2. Skillnaden var den att några av uppgifterna tagits bort då ändringar i användargränssnittet gjorde att vissa uppgifter inte längre var relevanta att utföra. Exempelvis behöver användaren i det nya gränssnittet inte kopiera föregående periods projekt då dessa nu är "defaultvärden". Attitydformuläret såg också i stort ut som det som användes vid undersökning 2. Även här togs ett antal frågor bort då dessa inte kunde besvaras p.g.a. att vi i denna undersökning inte hade ett riktigt system. Exempelvis togs frågor bort som berörde felmeddelanden och hur lätt systemet var att lära sig.

Observationen i denna undersökning visade att det fortfarande fanns vissa användbarhetsproblem i användargränssnittet men dessa var dock inte lika allvarliga som i undersökning 2. Det problem som flera av användarna påpekade i denna undersökning var att det saknades information i användargränssnittet. Detta gällde främst ett sökfönster där exempelvis två användare ville ha en extra kolumn i sökresultatet. Vidare ville två användare ha fler sökkriterier i samtliga sökfönster. För tillfället finns två sökkriterier vilket enligt användarna var för lite.

Resultatet från observationen visar också att användarna i vissa fall fick leta i menyerna för att hitta det de sökte efter. Detta var dock i en mycket liten skala med tanke på att dessa användare aldrig tidigare sett användargränssnittet och att de var vana vid den svenska versionen av Systemet. Vi konstaterar därför att termerna i menyerna har blivit mer begripliga och att menyens struktur har blivit mer logisk.

Några av användarna menade att anledningen till att de gick in på fel meny berodde på att de vant sig vid placeringen av funktionerna i det befintliga systemet. De gick in på samma meny på pappersprototypen som de brukar göra i det befintliga systemet för att hitta en specifik funktion. Tre av fyra användare ville t ex att funktionen "New

Time Sheet” skulle ligga under menyn ”Edit” då så är fallet i det befintliga systemet. Vi hade dock placerat denna funktion under ”File” för att på så sätt följa Windowsstandarden.

En av användarna var nybörjare på Systemet. Denne hade endast arbetat ett par veckor på företaget och hade inte hunnit lära sig Systemet ännu. Denna användare hittade informationen mycket enkelt i användargränssnittsprototypen vilket visar på en logisk placering och lättförståeliga begrepp. Detta visade tydligt på att användbarheten ökat i användargränssnittet.

Observationen och diskussionen kring användargränssnittet visade även att användarna tyckte det var lätt att hämta in information till huvudfönstret. Dels med hjälp av ”pull-down”-menyerna i den undre delen av tidsarket, vilket var en förändring från det befintliga systemet, och dels genom sökfönstren. Undersökningen visade även att användarna var mycket positiva till utformningen av sökfönstren. De ansåg att det var enkelt och smidigt att söka, men de påpekade dock att det saknades en del information i fönstren.

Alla användare i undersökningen var positiva till veckoalmanackan som vi placerat på vänsterkanten i huvudfönstret. De förstod genast hur den skulle användas. Det kan även vara värt att kommentera att knappen ”Submit Time Sheet”, som är en funktion som skickar iväg veckans tidsark för att godkännas, hittades direkt av användarna, trots det engelska begreppet. En av användarna ansåg dock att placeringen av denna knapp inte var helt logisk.

Resultatet från attitydformuläret såg ut på följande sätt:

- | |
|--|
| <ol style="list-style-type: none">1. Användargränssnittet har fula färger (1,0)2. Vägen till önskad information är inte självklar (3,5)3. Informationen är ej logiskt placerad (4,0)4. Användning av termer i prototyperna är inkonsekvent (4,0)5. Användargränssnittet är svårt att överskåda (4,25)6. Det finns för mycket överflödig information på prototyperna (4,25)7. Att korrigera ett misstag är svårt (4,25) |
|--|

Figur 7.13. Resultat från attitydformuläret i undersökning 4

Siffrorna i parentesen i figuren ovan är medelvärde av svaret. Värdena ligger mellan 1 och 5, där 1 är negativt och 5 är positivt. Vi vill påpeka att dessa värden är mycket ungefärliga och bara ger en känsla över användarnas attityd till användargränssnittet.

Som nämnts tidigare har vi valt att inte göra prototyperna i färg varför detta fortfarande är ett stort användbarhetsproblem. Vi kommer i fortsättningen inte att ta upp detta som ett användbarhetsproblem i resultatet.

Tittar vi vidare på resultatet i figuren ovan ser vi att resterande värden är höga, varför dessa inte kan ses som några allvarliga användbarhetsproblem. Hur som helst visar resultatet att vägen till önskad information i användargränssnittet inte alltid var självklar för användarna. Användarna ansåg även att informationen var något ologiskt placerad i användargränssnittet och att termerna kunde vara något inkonsekventa.

Vi kan dock, som sagt, konstatera att medelvärdet för dessa, och resterande användbarhetsproblem i figuren ovan är mycket höga, särskilt i jämförelse med resultatet från attitydformuläret i undersökning 2. Detta visar att användarna är mer positiva till det nya användargränssnittet än till användargränssnittet i det befintliga systemet, vilket kan ses som att användbarheten i systemet har ökat.

8. Diskussion

Vi har valt att dela upp diskussionen i två avsnitt, Slutligt resultat och Reflektion. I det slutliga resultatet diskuterar vi resultaten från våra undersökningar utifrån vår problemställning. I reflektionen diskuterar vi ämnet i ett bredare perspektiv och tar upp problem och kritik till vår studie.

8.1 Slutligt resultat

I det här avsnittet sammanfogar vi resultatet från våra fyra undersökningar för att på så sätt svara på vår problemställning. Då vår problemformulering är indelad i två underfrågor har vi valt att även dela upp redovisningen av det slutliga resultatet i två delar: Resultatet skall dels beskriva den metod som vi tagit fram som skall ligga till grund för utvecklingen av användargränssnittet i ett befintligt system, och dels skall den besvara frågan om användbarheten ökat i användargränssnittet genom användningen av denna metod.

- *Kan man genom att integrera användaren i användargränssnittsdesignen, öka användbarheten i ett system?*

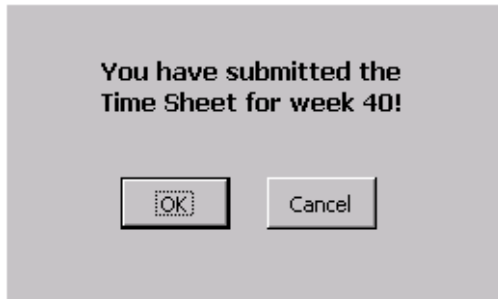
Svaret på denna fråga fick vi genom att utvärdera både det befintliga systemets användargränssnitt (undersökning 2) och den framtagna prototypen (undersökning 4) på liknande sätt. Genom att jämföra resultaten från undersökning 2 och undersökning 4 kan vi mäta om användargränssnittets användbarhet ökat eller ej.

Vi börjar med att göra en jämförelse mellan resultatet från observationerna i undersökning 2 och undersökning 4. Därefter diskuterar vi den avslutande intervjufrågan i undersökning 2 och jämför denna med användarnas synpunkter i undersökning 4. Slutligen jämförs resultaten från attitydformulären i de båda undersökningarna.

Det största användbarhetsproblemet från observationen i undersökning 2 var att **användarna fick leta i menyerna för att hitta funktionerna**. Detta problem återfanns även i undersökning 4 men inte i samma utsträckning. Orsaken till att problemet fanns kvar kan bero på att användaren till stor del, vilket nämns under resultatet i undersökning 4, vant sig vid det befintliga systemet. De letade under samma meny i det nya användargränssnittet som de brukade göra i användargränssnittet i det befintliga systemet och tänkte inte på logiken i menynamnen.

Det andra användbarhetsproblemet från observationen i undersökning 2 var att **användaren kände sig osäker på om funktionen utförde det som användaren trodde att den skulle göra**. I utformningen av prototypen har vi varit noggranna med att ge användaren feedback på allt som denne gör. Text när användaren är klar med sin tidsredovisning för veckan skall tidsarket godkännas och skickas iväg. I det befintliga systemet saknas feedback för denna aktivitet, vilket gör användaren osäker

på om händelsen har utförts eller ej. I prototypen lade vi in feedback för detta, se figur 8.1. nedan. Denna feedback tyckte användarna i undersökning 4 var mycket bra. Alla användare i undersökning 4 var även mycket positiva till veckoalmanackan i användargränssnittets prototypens huvudfönster. Även denna ger feedback i form av att "Submit"-kolumnen i veckoalmanackan ändras från "No" till "Yes" när användaren godkännt och skickat iväg veckans tidsark.



Figur 8.1. Exempel på feedback i det nya användargränssnittet

Resultatet från observationen i undersökning 2 visade även på att **användarna inte förstod hur de skulle använda funktionerna och att de inte hittade dem i Systemet**. Användarna i undersökning 4 hade inga problem att förstå hur funktionerna på pappersanvändargränssnittet fungerade. De hittade också funktionerna utan några större problem.

Både resultatet från observationen och den avslutande intervjun i undersökning 2 visade att användargränssnittet använde **ologiska termer på menyer och i övriga delar av systemet**. Det var ingen av användarna i undersökning 4 som nämnde något om ologiska termer. De nämnde att det var lite ovant med engelskan. Vi anser dock att engelskan inte utgjorde något större problem för användarna då de utan någon större svårighet fann det de letade efter.

Ett annat användbarhetsproblem som användaren uttryckte det i den avslutande intervjun i undersökning 2 var att **Systemet hade mycket onödigt information**. I utformningen av användargränssnittsprototyperna tog vi bort information som vi tillsammans med användarna ansåg vara orelevant. Undersökning 4 visade dock att viss information saknades i prototypen, och då främst i sökfönstren.

Vidare ansåg användarna i undersökning 2 att **informationen i menyerna hade en ologisk placering**. I utvärderingen av prototyperna fick några användare leta i menyerna för att finna det de sökte efter. Men som tidigare nämnts berodde detta till stor del på att de automatiskt gick in på den menyn där funktionen fanns i det befintliga systemet.

Det sista problemet som användarna påpekade med Systemet i undersökning 2 var att **de ville ha fler "defaultvärden"**. Användargränssnittsprototypen har betydligt fler "defaultvärden" än det befintliga systemet. Exempelvis har vi lagt in föregående periods projekt som "defaultvärde" för den nästföljande perioden. Vidare kommer användaren, efter inloggningen, alltid till tidsarket för den aktuella veckan, till skillnad mot det befintliga systemet där man kommer till tidsarket för den första veckan.

Går vi vidare och jämför resultaten från attitydformulären i de båda undersökningarna (se figuren nedan) ser vi att användarnas attityd till det nya användargränssnittet (den högra rutan, undersökning 4) är mer positiva i jämförelse med det gamla (den vänstra rutan, undersökning 2). Det första användbarhetsproblemet i figuren nedan som handlar om användargränssnittets färger kommer vi, som tidigare nämnts, inte vidare att diskutera.

1. Användargränssnittet har fula färger (1,25)	1. Användargränssnittet har fula färger (1,0)
2. Användargränssnittet är svårt att överskåda (1,5)	2. Vägen till önskad information är inte självklar (3,5)
3. Felmeddelanden klargör inte problemet och ger förslag på lösning (1,5)	3. Informationen är ej logiskt placerad (4,0)
4. Vägen till önskad information är inte självklar (1,66)	4. Användning av termer i prototyperna är inkonsekvent (4,0)
5. Att korrigera ett misstag är svårt (1,83)	5. Användargränssnittet är svårt att överskåda (4,25)
6. Att använda sökfunktionen är svårt (2,0)	6. Det finns för mycket överflödig information på prototyperna (4,25)
	7. Att korrigera ett misstag är svårt (4,25)

Figur 8.2. Sammanställning av resultaten från attitydformulären i undersökning 2 respektive undersökning 4

Det andra användbarhetsproblemet som attitydformuläret i undersökning 2 visade, var att **användargränssnittet är svårt att överskåda**. Detta användbarhetsproblem återfanns även i undersökning 4. Men tittar man på siffran i parentesen som är medelvärde av användarnas svar, där 1 är negativt och 5 är positivt, ser man att problemet har minskat radikalt i undersökning 4.

Vägen till önskad information har också förbättrats i undersökning 4 i jämförelse med undersökning 2. Problemet att korrigera misstag är inte heller ett stort problem i det nya användargränssnittet.

Det bör nämnas att vissa attitydfrågor tagits bort i undersökning 4, vilket har gjort att vi inte kan jämföra alla svaren i attitydformulären från de två undersökningarna.

Avslutningsvis anser vi att vi har besvarat den första underfrågan i vår problemställning nämligen: *kan man genom att integrera användaren i användargränssnittsdesignen, öka användbarheten i ett system?* Svaret på frågan är ”ja”. Genom att utvärdera användargränssnittet före och efter förändringarna har vi fått fram att användbarheten i användargränssnittet ökat avsevärt.

- ***Hur skall den metod se ut som ligger till grund för utvecklingen av användargränssnittet i ett befintligt system?***

Då svaret på den första frågan visade sig vara positivt anser vi att den metod som vi arbetade efter är bra att följa för att öka användbarheten i ett befintligt system. Nedan beskriver vi stegen i denna metod. Detta är generella riktlinjer som man bör kunna applicera på de flesta system som kännetecknas av dålig användbarhet.

1. Utvärdera det befintliga användargränssnittet utan användare

- Använd en Heuristisk Evaluering och Standard Inspektion för att finna användbarhetsproblemen i systemet.
- Utgå från de tio Heuristiska tumreglerna samt teorins generella riktlinjer för hur man utformar ett användbart användargränssnitt.

- Försök att ha flera utvärderare som oberoende av varandra utvärderar systemet. Detta för att finna så många användbarhetsproblem som möjligt. Fem utvärderare anses vara lagom enligt Nielsen¹²⁷, men en utvärderare är dock bättre än ingen alls.
 - Vid sammanställning av resultatet, försök ta fram de allvarligaste användbarhetsproblemen i systemet, dvs områden där systemet avviker markant från de Heuristiska tumreglerna.
2. *Utvärdera det befintliga användargränssnittet med användare*
- Kontakta målgruppen och fråga vilka som kan tänka sig att vara med i denna studie för att öka användbarheten i systemet.
 - Använd etnografisk observation tillsammans med intervju och enkät.
 - Vid utvärderingen av ett system skall följande personer deltaga; användare, testledare och observatör. Användaren får ett antal uppgifter som denne skall utföra i systemet. Testledaren skall leda, ställa frågor och uppmuntra användaren att "tänka högt" under testets gång, men får samtidigt inte hjälpa användaren att lösa uppgifterna. Observatören skall vara helt tyst under observationen och anteckna det användaren gör och säger.
 - Välj ut olika typer av användare. Välj ut minst fem personer. Bestäm tid och plats med användaren och skicka ett välkomstbrev med information om vad testet innebär.
 - Förbered den etnografiska observationen genom att ta fram uppgifter som användaren skall göra i systemet. Uppgifterna skall vara formulerade så att användaren känner igen sig i sina arbetsuppgifter. De skall även belysa de områden som man utifrån den Heuristiska evalueringen fick fram som de allvarligaste användbarhetsproblemen. Låt användarna bara se en uppgift i taget och låt användaren själv läsa varje fråga i lugn och ro.
 - Var väl förberedd på de händelser i systemet som uppgifterna genererar för att på så sätt kunna ställa frågor under tiden som uppgifterna utförs.
 - Se till att användaren känner sig bekväm och avslappnad i situationen. Förklara innan testet startar att detta inte är ett test som mäter användarens kunskaper om systemet, utan ett test som mäter användbarheten i användargränssnittet, vilket innebär att problem som kan uppstå inte beror på testpersonen utan på användargränssnittet.
 - Låt användaren fylla i en enkät i form av ett attitydformulär där frågornas svarsalternativ består av en skala från 1-5. Detta för att mäta vad användaren tycker om systemet.
 - Avsluta testet med en intervju där testpersonen får ge sina synpunkter på systemet och ge förslag på eventuella förbättringar.
3. *Utforma en prototyp av användargränssnittet med hjälp av "Quick and dirty ethnography"*
- Utforma först en prototyp utan användare utifrån de användbarhetsproblem som tagits fram i utvärderingarna av det befintliga användargränssnittet. Prototypen skall göras på papper och skall inte vara alltför noggrant gjord då detta kan hämma användaren att kladda och ändra på prototypen.

¹²⁷ Nielsen, Jakob, *Usability engineering* (1993)

- Välj ut minst tre grupper av användare med två användare i varje grupp. Varje grupp skall helst ha olika arbetsuppgifter i verksamheten. De två som skall jobba ihop under testet skall helst känna varandra.
- Utforma ett antal uppgifter som användarna skall utföra i användargränssnittet. Dessa uppgifter bör testa användbarheten i gränssnittet i så stor utsträckning som möjligt, och då främst de delar som ändrats från gränssnittet i det befintliga systemet. Uppgifterna ställs muntligt.
- Testet skall ha två testledare varav en har som uppgift att läsa uppgifterna och diskutera användargränssnittet med användarna. Den andra testledaren skall "agera dator" och flytta omkring delar av gränssnittet efter användarnas kommandon. Denne testledare behöver inte vara tyst i samma utsträckning som under observationen i den föregående utvärderingen, utan får gärna ställa frågor och kommentera händelser som den första testledaren eventuellt missar.
- Sammanställ resultatet från mötet med användarna så snart som möjligt efter testet då allt är färskt i minnet. Gå igenom användarnas synpunkter och förslag på förändringar i användargränssnittet och besluta vad som skall tas med i omdesignen av pappersprototypen.
- Omforma prototypen och testa denna på en ny grupp av användare.
- Fortsätt med denna process tills dess att en tillfredsställande prototyp har tagits fram. Man bör träffa minst tre grupper, men vid behov, gärna fler.

4. *Utvärdera den slutliga prototypen med användare*

- Gör om den handgjorda slutprototypen till en programmerad datorprototyp. Skriv därefter ut användargränssnittsprototypen på papper och utvärderar denna.
- Utvärdera användargränssnittsprototypen på samma sätt som utvärderingen av det befintliga systemets användargränssnitt med användare. Dvs välj ut ett antal användare (4-5 stycken) och låt dem lösa samma uppgifter som användarna fick göra vid utvärderingen av det befintliga systemet. Använd även samma attitydformulär. Eventuellt måste uppgifterna och attitydformuläret revideras en aning p.g.a. de ändringar som gjorts i systemet. Försök dock i möjligaste mån att ha kvar samma uppgifter och attitydfrågor för att lättare kunna jämföra resultaten.
- Om utvärderingen av användargränssnittet skulle visa på nya användbarhetsproblem skall dessa åtgärdas innan prototypen används som det slutliga användargränssnittet i systemet.

8.2 Reflektion

Denna uppsats har handlat om hur man kan öka användbarheten i ett befintligt system. Vi har valt att endast fokusera på användargränssnittet och hur detta skall utformas för nå en hög grad av användbarhet. Vi har i denna studie utvärderat ett befintligt systems användargränssnitt som kännetecknas av låg användbarhet och tillsammans med användaren tagit fram ett nytt, tillfredsställande användargränssnitt. Det nya användargränssnittet har i sin tur utvärderats för att på så sätt kunna mäta om användbarheten ökat i systemet. Vi har valt att stanna här. Man skulle kunna arbeta vidare med detta i form av att åtgärda de enstaka användbarhetsproblem i användargränssnittet som uppkom i utvärderingen av det nya användargränssnittet för att på så sätt ytterligare öka användbarheten. Vidare skulle man kunna undersöka

möjligheterna att realisera användargränssnittet i Systemet. Detta är dock ett stort arbete då man måste gå in och förstå programkoden som ligger bakom Systemet.

Ett förslag till vidare studier inom detta område är att undersöka huruvida det alltid är fördelaktigt att förbättra ett användargränssnitt i ett befintligt system eller om det ibland kan vara bättre att bygga ett helt nytt system. Detta kan t ex bero på hur den bakomliggande programkoden ser ut. Om denna kod är strukturerad och tydlig kan det vara enkelt att göra ändringar. Ett system som däremot kännetecknas av ostrukturerad och svårtolkad kod kan vara fördelaktigt att helt byta ut. Vi anser dock att vår metod kan användas oavsett om man vill vidareutveckla ett befintligt system eller om man väljer att bygga ett helt nytt system. Väljer man att bygga ett nytt system kan man med fördel använda vår metod för att utvärdera vilka problem som finns i dagens system för att på så sätt bygga ett system som bättre uppfyller användarnas behov.

Ett problem vi stötte på under vårt uppsatsskrivande var att Mind AB i Göteborg lade ner sin verksamhet innan vi var helt färdiga. P.g.a. denna händelse fick vi utföra den sista undersökningen på ett annat företag. Vi anser att vi trots detta problem fick ut en hel del av denna undersökning och att det gav oss ett jämförbart resultat. Vi har dock den uppfattningen att ett bättre och mer tillförlitligt resultat hade uppnåtts om även denna sista undersökning gjorts på Mind. Nackdelen med att göra den sista undersökningen på ett annat företag var att vi inte på samma sätt kände till dess verksamhet. På Mind hade vi varit under en längre tid och kände därför till både företaget och dess anställda. På det nya företaget var vi endast ett par timmar och hann aldrig på den tiden skapa en ordentlig bild av företaget. Användargränssnittet utvecklades i samarbete med de anställda på Mind och var därför anpassat till deras behov. Det är därför möjligt att resultatet blivit något annorlunda om alla undersökningar gjorts på ett och samma företag.

En annan nackdel med att göra den sista undersökningen på ett annat företag var att undersökningspersonerna inte kände oss vilket kan ha medfört att de inte vågade säga för mycket negativt om användargränssnittet då de kanske kände att de mer kritiserade oss som personer än gränssnittet.

Urvalet till den sista undersökningen blev i form av självselektion, eller rättare sagt de personer som var villiga att ställa upp den dagen vi var på företaget. Tyvärr gick inte urvalet att lösa på annat sätt då vi hade mycket begränsat med tid. P.g.a. detta kan urvalet ha blivit snett fördelat då kanske endast en typ av anställda blivit representerat i den sista undersökning.

Urvalet till undersökning 2 och 3 går också att diskutera. Urvalet baserades på det mail vi skickade ut där vi frågade vilka som ville och kunde ställa upp på våra undersökningar. Urvalet kan ha blivit snett då det kanske endast var en viss typ av människor som var villiga ställa upp. Hade vi gjort ett obundet slumpmässigt urval av hela populationen hade vi kanske fått ett urval som bättre stämde överens med Minds anställda.

Den metod vi har tagit fram och arbetat efter är en metod som vi själva har sammanställt utifrån teorin. Vi anser att den gav ett bra resultat men självklart går det att diskutera relevansen med de olika delarna i metoden. T ex är det frågan om

informationen vi fick i undersökning 1 var nödvändig för att göra undersökning 2. Vi anser dock att den Heuristiska Evalueringen gav oss en övergripande bild över vilka användbarhetsproblem Systemet hade och att den hjälpte oss att se vilka delar i Systemet vi skulle fokusera på under användbarhetstesterna. Samtidigt lärde vi oss Systemet vilket är en förutsättning för att kunna observera användarna i undersökning 2. En nackdel kan här vara att vi i undersökning 2 färgades av det vi kom fram till i undersökning 1, dvs att vi eventuellt fokuserade allt för mycket på de problem som vi själva kommit fram till.

Vidare kan man diskutera det attitydformulär som vi använde i undersökning 2 och 4 för att kunna mäta om användbarheten ökat eller ej i användargränssnittet. Detta formulär utformades tidigt i vår studie då vi inte hade tillräckligt kunskap inom området. Detta resulterade i att attitydformuläret saknade viss information. Vi upptäckte även att den typ av undersökning vi utförde, där vi observerade, intervjuade och diskuterade med användarna när de använde Systemet, krävde viss övning. Man skall exempelvis tänka på att inte hjälpa till, inte prata för mycket, men inte heller för lite, inte skratta vid fel tillfällen etc under undersökningens gång. Vi gjorde dock flera av dessa misstag i början och det krävdes flera användbarhetstester innan vi kände oss någorlunda säkra på hur vi skulle agera.

9. Slutsats

Utifrån det resultat vi fick fram kan vi dra slutsatsen att vår metod är en bra metod för att förbättra användbarheten hos ett befintligt system. Detta innebär att man utvärderar och omformar användargränssnittet tillsammans med användare. Om man själv inte är expert inom gränssnittsdesign, bör man även ta hjälp av experters riktlinjer för hur man utformar användargränssnitt.

Vi kan även dra slutsatsen att vår metod kan generaliseras, dvs användas på alla system som kännetecknas av dålig användbarhet. Vi anser också att ju bättre en gränssnittsutvecklare känner och förstår den verksamhet där systemet skall verka, desto större är möjligheten att systemet uppfyller användarnas behov.

Litteraturförteckning

Böcker:

- Blomberg, J et al, *Ethnographic Field Methods and Their Relation to Design Participatory Design: Principles and Practices* (1993)
- Easterby-Smith, Mark et al., *Management Research; An introduction*, Sage Publication Ltd (1991)
- Flynn, Donald, *Information System Requirements: Determination & Analysis* (1997)
- Fossum, Per, *Metoder för att hitta användbarhetsproblem hos datorsystem*, SISU (1996)
- Galitz, Wilbert, *The essential guide to user interface design*, John Wiley & Sons, inc (1996)
- Hix, Deborah & Hartson H.R, *Developing user interfaces: Ensuring usability through product and process*, John Wiley & Sons, inc (1993)
- Hughes, John et al., *Moving Out from the Control Room: Ethnography in System Design*, Lancaster University, Computing and Sociology Department (1994)
- Mayhew, Deborah, *The usability engineering lifecycle*, Morgan Kaufmann Publishers, inc (1999)
- Nielsen, J & Mack, R.L, *Usability Inspection Methods*. John Wiley & Sons Inc., 1994
- Nielsen, Jakob, *Usability engineering*, Academic Press, inc (1993)
- Patel, Runa & Davidsson, Bo, *Forskningsmetodikens grunder: Att planera, genomföra och rapportera en undersökning*, Studentlitteratur, Lund (1991)
- Preece, Jenny, *A guide to usability: Human factors in computing*, Addison-Wesley (1993)
- Preece, Jenny, *Human-computer-interaction*, Addison-Wesley (1996)
- Pressman, R.S. *Software Engineering: a practitioner's approach* (European edition) McGraw-Hill International (UK) Ltd. (1994)
- Rubin, Jeffrey, *Handbook of usability testing*, John Wiley & Sons, inc (1994)
- Shneiderman, Ben, *Designing the user interface strategies for effective human – computer – interaction*, Addison-Wesley (1998)

- Undheim, Johan Olav, *Statistik från ord till formel*, Studentlitteratur, Lund (1988)

Webbdokument:

- Department of computer science, University of Maryland, Usability engineering, www.cs.umd.edu/~zzj/UsabilityHome.html
- Jacob Nielsens hemsida, www.best.com/~jthom/usability/usable.htm
- Prototyping for Tiny Fingers, www.cs.mu.oz.au/~tetra/index1.html

Bilagor

Bilaga A: Undersökning 1

Bilaga A1: Användargränssnittet för tidsarket (Time sheet)

Employee Information

Employee No. 1084
Name Johanna Palm
Department Main Department

Details

Week 35
Year 2008
Date 09/04/2008 - 09/10/2008
Week Calendar No. 1

Total

	Date	Individual	Interval	Total	Round	OverTime
Monday	09/04/2008	8,0	8,0	8,0	8,0	0,0
Tuesday	09/05/2008	8,0	8,0	8,0	8,0	0,0
Wednesday	09/06/2008	8,0	8,0	8,0	8,0	0,0
Thursday	09/07/2008	8,0	8,0	8,0	8,0	0,0
Friday	09/08/2008	8,0	8,0	8,0	8,0	0,0
Saturday	09/09/2008	0,0	8,0	0,0	8,0	0,0
Sunday	09/10/2008	0,0	8,0	0,0	8,0	0,0
Total		40,0	8,0	40,0	48,0	0,0

Activities

Time Activity 1 0,0
Time Activity 2 0,0
Time Activity 3 0,0
Total Time Activities 0,0
Calculated OverTime 0,0
Balance 0,0

Status

Created by Johanna Palm
Date 10/10/2008
Submitted ☐
Submitted by Johanna Palm
Date 10/11/2008
Time Submitted ☐

Job No.	Job Name	Act. No.	Activity Name	Task	Task Description	Monday	Tuesday	Wednesday	Thursday	Friday
200815	Odette & Co Ts	101	Inspection - time	315	Customer Training	0,0	8,0	8,0	0,0	8,0
200818	Dilmar Postel	105	Launch - time	318	Project Meeting	0,0	8,0	8,0	0,0	8,0

Bilaga A2: Användargränssnittet för sökfönstret "Find Time Sheet"

Find Time Sheet

File Edit Columns Window

Columns: Employee No. = 1084

Employee No.	Start Date	Closing Date	Week No.	Part	Year	Approved	Approved by	Approval Date	Supervisor No.
jpa	09/04/2008	09/07/2008	35	A	2008	No			jpa
jpa	09/04/2008	09/07/2008	35	B	2008	No			jpa
jpa	09/04/2008	09/10/2008	36		2008	Yes	Astrida Öster	12/05/2008	jpa
jpa	09/11/2008	09/17/2008	37		2008	Yes	Johanna Pa	11/01/2008	jpa
jpa	09/18/2008	09/24/2008	39		2008	No			jpa
jpa	09/25/2008	09/28/2008	39	A	2008	No			jpa
jpa	10/01/2008	10/01/2008	39	B	2008	No			jpa
jpa	10/03/2008	10/08/2008	40		2008	Yes	Johanna Pa	10/04/2008	jpa
jpa	10/08/2008	10/15/2008	41		2008	Yes	Johanna Pa	10/04/2008	jpa
jpa	10/16/2008	10/22/2008	42		2008	Yes	Johanna Pa	10/04/2008	jpa
jpa	10/23/2008	10/29/2008	43		2008	Yes	Astrida Öster	10/25/2008	jpa
jpa	10/30/2008	10/31/2008	44	A	2008	Yes	Johanna Pa	11/02/2008	jpa
jpa	11/01/2008	11/06/2008	44	B	2008	Yes	Johanna Pa	10/27/2008	jpa
jpa	11/06/2008	11/13/2008	45		2008	Yes	Johanna Pa	11/02/2008	jpa
jpa	11/13/2008	11/18/2008	46		2008	Yes	Johanna Pa	11/03/2008	jpa
jpa	11/20/2008	11/26/2008	46	A	2008	No			jpa
jpa	12/01/2008	12/03/2008	48	B	2008	No			jpa
jpa	12/04/2008	12/10/2008	49		2008	No			jpa

Bilaga B: Undersökning 2

Bilaga B1: Uppgifter att utföra i systemet

Uppgift 1

- a) Jag vill tidsredovisa för den senaste perioden i Systemet (i det här fallet vecka 46). Jag har arbetat i samma projekt som föregående vecka. Vänta med att lägga in antal timmar på de olika projekten.
- b) Jag har även arbetat i ett nytt projekt som heter "Prenax – Version 1" och som jag därför vill lägga in i min projektlista. Activity number är 105 och Task number är 318 för detta projekt.
- c) Jag har även avslutat ett av mina projekt (välj vilket som helst) och vill därför ta bort det från projektlistan i mitt timesheet.
- d) Jag vill nu lägga in arbetstiden för de olika projekten. Jag arbetar heltid, dvs 40 timmar i veckan. Antal timmar på de olika projekten har ingen betydelse.
- e) Avsluta tidsredovisningen.

Uppgift 2

- a) Det kommer vara mycket att göra på jobbet nästföljande vecka varför jag vill förbereda min tidsredovisning. Jag skapar därför ett nytt timesheet för nästa tidsperiod och lägger in projektet med "job number" 300022.
- b) Jag ångrar mig och vill istället kopiera in förra veckans projekt i detta nyskapade timesheet.
- c) Nisse frågar mig om jag kan ta fram det "jobnumber" som projektet Everyday Webguide har då jag ändå är inne i Systemet.
- d) Då det har blivit missförstånd i faktureringen ringer Greta, som är ekonomiansvarig, och frågar vilka projekt jag arbetade i under vecka 36.
- e) Greta ber mig samtidigt kontrollera att timesheeten för mina hittills arbetade veckor är godkända då det har slarvats med detta den senaste tiden.
- f) Jag inser att jag inte kommer att arbeta med de inlagda projekten nästa vecka varför jag vill ta bort det senaste skapta timesheetet.

Bilaga B2: Attitydformulär

IS = Inget Svar

a) Övergripande och generella intryck till Systemet

Vad tycker du om Systemet?

Hemskt	1	2	3	4	5	IS	Underbart
Frustrerande	1	2	3	4	5	IS	Tillfredställande
Tråkig	1	2	3	4	5	IS	Stimulerande
Svårt	1	2	3	4	5	IS	Lätt
För lite information	1	2	3	4	5	IS	För mycket information
Fula färger	1	2	3	4	5	IS	Fina färger

b) Fönster

Användargränssnittet är lättöverskådligt	instämmer ej	instämmer
	1 2 3 4 5	IS
Information som visas i fönstren är relevant	instämmer ej	instämmer
	1 2 3 4 5	IS
Informationen är logisk placerad	instämmer ej	instämmer
	1 2 3 4 5	IS
Vägen till önskad information är självklar	aldrig	alltid
	1 2 3 4 5	IS

c) Terminologi och systeminformation

Användningen av termer i systemet är	inkonsekvent	konsekvent
	1 2 3 4 5	IS
Funktionerna i systemet uträttar det du tror att de ska uträtta	aldrig	alltid
	1 2 3 4 5	IS
Felmeddelanden är lätta att förstå	instämmer ej	instämmer
	1 2 3 4 5	IS
Felmeddelande klargör problemet och ger förslag på en lösning	instämmer ej	instämmer
	1 2 3 4 5	IS

d). Inläarning

Att lära sig Systemet var	svårt	lätt
	1 2 3 4 5	IS
Att komma ihåg vart något finns är	svårt	lätt
	1 2 3 4 5	IS
Antalet steg för att utföra en uppgift är lagom	instämmer ej	instämmer
	1 2 3 4 5	IS

e). Funktioner

Att korrigera ett misstag är	svårt	lätt
	1 2 3 4 5	IS
Att använda sökfunktionen är	svårt	ätt
	1 2 3 4 5	IS

Bilaga B3: Intervjufråga

Vad anser du är det största problemet med Systemets användargränssnitt?

- Finns det något du skulle vilja ändra på?

Bilaga C: Undersökning 3

Bilaga C1: Uppgifter att utföra på pappersprototypen

1. Lägga in ett nytt projekt i sin projektlista. Jobnamnet är SFS Extranet, Aktivitet nr är 105 och Task description är Project meeting.
2. Godkänn och skicka iväg timesheetet.
3. Ta fram timesheetet för vecka 31.
4. Kontrollera att dina hittills arbetade veckor är submittade. Kan du kontrollera detta någonstans?
5. Skapa ett nytt timesheet för nästa period
6. Ta bort en rad i din projektlista

Bilaga C2: Diskussion

Time Sheet

- Vänstermenyns delar
 - kalendern
 - godkänn timesheet
- Menyn
 - är funktioner logiskt placerade i menyn?
 - är det logiska termer i menyn och dess undermenyer?
- Hela fönstret
 - relevant information?
 - är informationen på rätt ställe i fönstret?

Find Timesheet

- Är sökfunktionen förstålig?
- Hela fönstret
 - relevant information?
 - är informationen på rätt ställe i fönstret?
 - är knappar och fält bra placerade i fönstret?
 - diskutera kolumner, vilka ska vara med?, vilka saknas?

Dessa frågor diskuteras även i sökfönstren Find Job, Find Activity och Find Task

Bilaga C3: Prototyp med ändringar från pilottest

Time Sheet
File Edit Find Submit Help

Find Job
~~Find Time Sheet~~

Employee Information
Employee No. _____
Name _____
Department Main Department
Supervisor _____

Period
Week 49
Date 2000-12-04 - 2000-12-10

Week
29 Yes
30 Yes
31 Yes
32 Yes
33 Yes
34 Yes
35 Yes
36 Yes
37 Yes
38 Yes
39 Yes
40 Yes
41 Yes
42 Yes
43 Yes
44 Yes
45 Yes
46 Yes
47 Yes
48 Yes
49 Yes
50 Yes
51 Yes
52 Yes

Submit
Time Sheet: 186

Time	Date	External	Internal	Total	Fixed	Overtime
Monday	2000-12-04	0,0			8,0	
Tuesday	2000-12-05	0,0			8,0	
Wednesday		0,0			8,0	
Thursday						
Friday						
Saturday						
Sunday						
Total						

Job No	Job Name	Act No	Activity Name	Task No	Task Description	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Remarks
10002	Code-13panda uppt	105	Launch-time	318	Project Meeting								
10003	JFS Exchange	105	Launch-time	318	Project Meeting								

Handwritten notes in red:
10002 Code-13panda uppt
10003 JFS Exchange
Project Meeting

Bilaga C4: Prototyp med ändringar från första användargruppen

Time Sheet
File Edit Find Submit Help

Find Job

Employee Information
Employee No. _____
Name _____
Department Main Department
Supervisor _____

Period
Week 49
Date 2000-12-04 - 2000-12-10

Week
29 Yes
30 Yes
31 Yes
32 Yes
33 Yes
34 Yes
35 Yes
36 Yes
37 Yes
38 Yes
39 Yes
40 Yes
41 Yes
42 Yes
43 Yes
44 Yes
45 Yes
46 Yes
47 Yes
48 Yes
49 Yes
50 Yes
51 Yes
52 Yes

Submit
Time Sheet

Time	Date	External	Internal	Total	Fixed	Overtime
Monday	2000-12-04	0,0			8,0	
Tuesday	2000-12-05	0,0			8,0	
Wednesday		0,0			8,0	
Thursday						
Friday						
Saturday						
Sunday						
Total						

Job No	Job Name	Act No	Activity Name	Task No	Task Description	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Remarks
10002	Code-13panda uppt	105	Launch-time	318	Project Meeting								
10003	JFS Exchange	105	Launch-time	318	Project Meeting								

Bilaga C5: Prototyp med ändringar från tredje användargruppen

Time Sheet

File Edit Find Help

Week	Submitted
20	Yes
21	Yes
22	Yes
23	Yes
24	Yes
25	Yes
26	Yes
27	Yes
28	Yes
29	Yes
30	Yes
31	Yes
32	Yes
33	Yes
34	Yes
35	Yes
36	Yes
37	Yes
38	Yes
39	Yes
40	Yes
41	Yes
42	Yes
43	Yes
44	Yes
45	Yes
46	Yes
47	Yes
48	Yes
49	Yes
50	Yes
51	Yes
52	Yes
53	Yes
54	Yes
55	Yes
56	Yes
57	Yes
58	Yes
59	Yes
60	Yes
61	Yes
62	Yes
63	Yes
64	Yes
65	Yes
66	Yes
67	Yes
68	Yes
69	Yes
70	Yes
71	Yes
72	Yes
73	Yes
74	Yes
75	Yes
76	Yes
77	Yes
78	Yes
79	Yes
80	Yes
81	Yes
82	Yes
83	Yes
84	Yes
85	Yes
86	Yes
87	Yes
88	Yes
89	Yes
90	Yes
91	Yes
92	Yes
93	Yes
94	Yes
95	Yes
96	Yes
97	Yes
98	Yes
99	Yes
100	Yes

Employee Information

Employee No. _____
 Name _____
 Department Main Department
 Supervisor _____

Period

Week 99
 Date 2000-12-04 - 2000-12-10

Time	Date	External	Internal	Total	Fixed	Overtime
Monday	2000-12-04	0.0			8.0	
Tuesday	2000-12-05	0.0			8.0	
Wednesday		0.0			8.0	
Thursday						
Friday						
Saturday						
Sunday						
Total						

Submit Time Sheet

Bilaga C6: Slutlig användargränssnittsprototyp för tidsarket (Time Sheet)

Time Sheet																																																																																
File		Edit	Find	Help																																																																												
Week:	Submitted		Employee Information				Period																																																																									
29	Yes		Employee No: 691202				Week: 41																																																																									
30	Yes		Name: Greta Galbo				Date: 2001-01-01 - 2001-01-07																																																																									
31	Yes		Department: Main Department																																																																													
32	Yes		Supervisor: Hulkun																																																																													
33	Yes																																																																															
34	Yes																																																																															
35	Yes																																																																															
36	Yes																																																																															
37	Yes																																																																															
38	Yes																																																																															
39	Yes																																																																															
40	Yes																																																																															
41	No																																																																															
<input type="button" value="Submit Time Sheet"/>																																																																																
<table border="1"> <thead> <tr> <th colspan="2">Time</th> <th>Date</th> <th>External</th> <th>Internal</th> <th>Total</th> <th>Fixed</th> <th>Overtime</th> </tr> </thead> <tbody> <tr><td>Monday</td><td></td><td>20001-01-01</td><td>0.0</td><td>0.0</td><td>0.0</td><td>8.0</td><td>0.0</td></tr> <tr><td>Tuesday</td><td></td><td>20001-01-02</td><td>0.0</td><td>0.0</td><td>0.0</td><td>8.0</td><td>0.0</td></tr> <tr><td>Wednesday</td><td></td><td>20001-01-03</td><td>0.0</td><td>0.0</td><td>0.0</td><td>8.0</td><td>0.0</td></tr> <tr><td>Thursday</td><td></td><td>20001-01-04</td><td>0.0</td><td>0.0</td><td>0.0</td><td>8.0</td><td>0.0</td></tr> <tr><td>Friday</td><td></td><td>20001-01-05</td><td>0.0</td><td>0.0</td><td>0.0</td><td>8.0</td><td>0.0</td></tr> <tr><td>Saturday</td><td></td><td>20001-01-06</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td></tr> <tr><td>Sunday</td><td></td><td>20001-01-07</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td><td>0.0</td></tr> <tr><td>Total</td><td></td><td></td><td>0.0</td><td>0.0</td><td>0.0</td><td>40.0</td><td>0.0</td></tr> </tbody> </table>									Time		Date	External	Internal	Total	Fixed	Overtime	Monday		20001-01-01	0.0	0.0	0.0	8.0	0.0	Tuesday		20001-01-02	0.0	0.0	0.0	8.0	0.0	Wednesday		20001-01-03	0.0	0.0	0.0	8.0	0.0	Thursday		20001-01-04	0.0	0.0	0.0	8.0	0.0	Friday		20001-01-05	0.0	0.0	0.0	8.0	0.0	Saturday		20001-01-06	0.0	0.0	0.0	0.0	0.0	Sunday		20001-01-07	0.0	0.0	0.0	0.0	0.0	Total			0.0	0.0	0.0	40.0	0.0
Time		Date	External	Internal	Total	Fixed	Overtime																																																																									
Monday		20001-01-01	0.0	0.0	0.0	8.0	0.0																																																																									
Tuesday		20001-01-02	0.0	0.0	0.0	8.0	0.0																																																																									
Wednesday		20001-01-03	0.0	0.0	0.0	8.0	0.0																																																																									
Thursday		20001-01-04	0.0	0.0	0.0	8.0	0.0																																																																									
Friday		20001-01-05	0.0	0.0	0.0	8.0	0.0																																																																									
Saturday		20001-01-06	0.0	0.0	0.0	0.0	0.0																																																																									
Sunday		20001-01-07	0.0	0.0	0.0	0.0	0.0																																																																									
Total			0.0	0.0	0.0	40.0	0.0																																																																									
<table border="1"> <thead> <tr> <th>Job No.</th> <th>Job Name</th> <th>Act No.</th> <th>Activity Name</th> <th>Task No.</th> <th>Task Description</th> <th>Mon</th> <th>Tue</th> <th>Wed</th> <th>Thu</th> <th>Fri</th> <th>Sat</th> <th>Sun</th> <th>Remarks</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>300012</td> <td>RoadB</td> <td>104</td> <td>Reclamation</td> <td>303</td> <td>Project Meetr</td> <td>4</td> <td>4</td> <td>4</td> <td>4</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2</td> <td>300014</td> <td>Enahor</td> <td>105</td> <td>Customtalk</td> <td>303</td> <td>Project Meetr</td> <td>4</td> <td>4</td> <td>4</td> <td>4</td> <td></td> <td></td> <td></td> </tr> <tr> <td>3</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>									Job No.	Job Name	Act No.	Activity Name	Task No.	Task Description	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Remarks	1	300012	RoadB	104	Reclamation	303	Project Meetr	4	4	4	4				2	300014	Enahor	105	Customtalk	303	Project Meetr	4	4	4	4				3														4															
Job No.	Job Name	Act No.	Activity Name	Task No.	Task Description	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Remarks																																																																			
1	300012	RoadB	104	Reclamation	303	Project Meetr	4	4	4	4																																																																						
2	300014	Enahor	105	Customtalk	303	Project Meetr	4	4	4	4																																																																						
3																																																																																
4																																																																																

Bilaga C7: Pull-down menyer i tidsarket

File	Edit	Find	Help
New Time Sheet	Undo	Find Job	About Maconomy
Print	Cut	Find Activity	
Close	Copy	Find Task	
	Paste		
	Delete	Find Time Sheet	
	Delete Time Sheet	Find Previous Time Sheet	
	Insert Time Sheet Line	Find Next Time Sheet	
	Submit Time Sheet		

Bilaga C8: Slutlig användargränssnittsprototyp för sökfönstret "Find Job"

Find Job

BACK CANCEL

Search for Job:

Job No:

Job Name:

FIND

Your Choices:

Job No:

Activity No:

Task No:

OK

	Job No	Job Name	Job Group	Blocked
1	30002	Enator	External Job	No
2	30046	Entra AB	External Job	No
3	30003	Evaholm AB	External Job	No
4	30004	Frolunda IT	External Job	No
5	30005	Gärlab	External Job	No
6	30006	Göinge ITCenter	External Job	No
7	30007	Härsberga Solution	External Job	No
8	30008	Hilleröd AB	External Job	No

Bilaga C9: Slutlig användargränssnittsprototyp för sökfönstret "Find Activity"

Find Activity

BACK CANCEL

Search for Activity:

Activity No:

Activity Name:

FIND

Your Choices:

Job No:

Activity No:

Task No:

OK

	Act No	Activity Name	Activity Type	Invoice	Blocked
1	101	Customtalk	Time	Yes	No
2	102	Definition-time	Time	Yes	No
3	103	Transition	Time	Yes	No
4	104	Launch	Time	Yes	No
5	105	Conclusion	Time	Yes	No
6	107	Inception	Amount	Yes	No
7	108	Definition-amount	Amount	Yes	No
8	109	Realization-amount	Amount	Yes	Yes

Bilaga C10: Slutlig användargränssnittsprototyp för sökfönstret ”Find Task”

Find Task

BACK

CANCEL

Search for Task:

Task No:

Task Description:

FIND

Your Choices:

Job No:

30002

Enter

Activity No:

102

Definition time

Task No:

OK

	Task No	Task Description	Blocked	
1	301	Projekt Management	No	
2	302	Production	No	
3	303	Test	No	
4	304	System Administration	No	
5	305	System Documentation	No	
6	306	Graphic Production	No	
7	307	Web Development	No	
8	308	Quality Audit	Yes	

Bilaga D: Undersökning 4

Bilaga D1: Uppgifter att lösa på prototyperna

Uppgift 1

a) Som Timesheetet visar arbetat jag med två projekt förra veckan, nämligen Ericsson och Rottinge. Dessa har jag även arbetat med denna veckan. Denna veckan har även arbetat i ett nytt projekt som heter "Enator" och som jag därför vill lägga in i min projektlista. Activity number är 102 och Task number är 301 för detta projekt.

b) Jag har även avslutat ett av mina projekt (välj vilket som helst) och vill därför ta bort det från projektlistan i mitt timesheet.

c) Jag vill nu lägga in arbetstiden för de olika projekten. Jag arbetar heltid, dvs 40 timmar i veckan. Antal timmar på de olika projekten har ingen betydelse.

d) Avsluta tidsredovisningen.

Uppgift 2

a) Det kommer vara mycket att göra på jobbet nästföljande vecka varför jag vill förbereda min tidsredovisning. Jag skapar därför ett nytt timesheet för nästa tidsperiod, dvs vecka 41.

b) Nisse frågar mig om jag kan ta fram det "jobnumber" som projektet "Rör AB" har då jag ändå är inne i Systemet.

c) Då det har blivit missförstånd i faktureringen ringer Greta, som är ekonomiansvarig, och frågar vilka projekt jag arbetade i under vecka 36.

d) Greta ber mig samtidigt kontrollera att timesheeten för mina hittills arbetade veckor är godkända då det har slarvats med detta den senaste tiden.

Bilaga D2: Attitydformulär

IS = Inget Svar

a) Övergripande och generella intryck till Systemet

Vad tycker du om Systemet?

Hemskt	1	2	3	4	5	IS	Underbart
Frustrerande	1	2	3	4	5	IS	Tillfredställande
Tråkig	1	2	3	4	5	IS	Stimulerande
Svårt	1	2	3	4	5	IS	Lätt
För lite information	1	2	3	4	5	IS	För mycket information
Fula färger	1	2	3	4	5	IS	Fina färger

b) Fönster

Användargränssnittet är lättöverskådligt	instämmer ej	instämmer
	1 2 3 4 5	IS
Information som visas i fönstren är relevant	instämmer ej	instämmer
	1 2 3 4 5	IS
Informationen är logisk placerad	instämmer ej	instämmer
	1 2 3 4 5	IS
Vägen till önskad information är självklar	aldrig	alltid
	1 2 3 4 5	IS

c) Terminologi och systeminformation

Användningen av termer i systemet är	inkonsekvent	konsekvent
	1 2 3 4 5	IS
Funktionerna i systemet utgör det du tror att de ska utgöra	aldrig	alltid
	1 2 3 4 5	IS

d). Inläring

Att komma ihåg vart något finns är	svårt	lätt
	1 2 3 4 5	IS
Antalet steg för att utföra en uppgift är lagom	instämmer ej	instämmer
	1 2 3 4 5	IS

e). Funktioner

Att korrigera ett misstag är	svårt	lätt
	1 2 3 4 5	IS
Att använda sökfunktionen är	svårt	lätt
	1 2 3 4 5	IS